Abstract

항목 추천은 일련의 항목(예: 웹사이트, 영화, 제품)에 대해 개인화된 순위를 예측하는 작업입니다.

(예: 웹사이트, 영화, 제품)에 대한 개인화된 순위를 예측하는 작업입니다.

이 문서에서는 암시적 피드백(예: 클릭, 구매)이 있는 가장 일반적인 시나리오를 살펴봅니다.

암시적 피드백을 통한 아이템 추천에는 행렬 인수분해(MF) 또는 적응형 인접한 이웃(kNN)과 같은 다양한 방법이 있습니다. 이러한 방법들은 개인화된 랭킹이라는 아이템 예측 작업을 위해 설계되었지만, 랭킹에 직접적으로 최적화된 방법은 없습니다. 이 논문에서는 문제에 대한 베이지안 분석에서 도출된 최대 사후 추정치인 개인화된 순위를 위한 일반적인 최적화 기준 BPR-Opt를 제시합니다. 또한 BPR-Opt와 관련하여 모델을 최적화하기 위한 일반적인 학습 알고리즘을 제공합니다. 이 학습 방법은 부트스트랩 샘플링을 사용한 확률적 경사 하강을 기반으로 합니다.

이 방법을 두 가지 최신 추천 모델에 적용하는 방법을 보여드립니다: 행렬 인수분해와 적응형 kNN에게 적용.

실험 결과, 개인화된 랭킹 작업에서 우리의 최적화 방법이 MF와 kNN의 표준 학습 기법을 능가하는 것으로 나타났습니다. 이 결과는 올바른 기준에 맞게 모델을 최적화하는 것이 중요하다는 것을 보여줍니다.

1.소개

콘텐츠 추천은 많은 정보 시스템에서 중요한 작업입니다. 예를 들어 Amazon과 같은 온라인 쇼핑 웹사이트는 각 고객에게 사용자가 관심을 가질 만한 제품을 맞춤 추천합니다.

다른 예로는 고객에게 영화를 추천하는 YouTube와 같은 비디오 포털이 있습니다. 개인화는 매출이나 조회수를 높일 수 있는 콘텐츠 제공업체와 흥미로운 콘텐츠를 더 쉽게 찾을 수 있는 고객 모두에게 매력적입니다. 이 문서에서는 아이템 추천에 초점을 맞춥니다. 항목 추천은 일련의 항목에 대한 사용자별 순위를 생성하는 작업입니다.

아이템에 대한 사용자의 선호도는 구매 내역, 시청 내역 등 사용자의 과거 시스템과의 상호 작용을 통해 학습됩니다.

가장 최근의 연구는 사용자가 평점과 같은 명시적인 피드백을 제공하는 시나리오에 관한 것입니다. 하지만 실제 시나리오에서는 대부분의 피드백이 명시적이지 않고 암시적입니다. 암시적 피드백은 클릭 수, 조회 시간, 구매 등을 모니터링하는 것처럼 자동으로 추적됩니다. 따라서 사용자가 자신의 취향을 명시적으로 표현할 필요가 없으므로 수집하기가 훨씬 쉽습니다. 사실 암시적 피드백은 거의 모든 정보 시스템에서 이미 사용할 수 있습니다(예: 웹 서버는 모든 페이지 액세스를 로그 파일에 기록합니다).

이 문서에서는 개인화된 랭킹을 위한 모델을 학습하는 일반적인 방법을 소개합니다. 이 작업의 기여는 다음과 같습니다:

1. 최적의 개인화 랭킹을 위한 최대 후행 추정자로부터 도출된 일반적인 최적화 기준 BPR-Opt를 제시합니다. BPR-Opt를 ROC 곡선 아래 영역의 최대화에 비유하여 보여줍니다.

2. BPR-Opt를 최대화하기 위해 훈련 트리플의 부트스트랩 샘플링을 통한 확률적 경사 하강을 기반으로 하는 일반 학습 알고리즘인 LearnBPR을 제안합니다.

이 알고리즘이 BPR-Opt 최적화를 위한 표준 경사 하강 기법보다 우수하다는 것을 보여줍니다.

3. 두 가지 최신 추천 모델 클래스에 LearnBPR을 적용하는 방법을 보여드립니다.

4. 실험을 통해 개인화된 랭킹 작업의 경우 BPR로 모델을 학습하는 것이 다른 학습 방법보다 성능이 뛰어나다는 것을 경험적으로 보여줍니다.

2. Related Work

추천 시스템에서 가장 많이 사용되는 모델은 k-최근접 이웃(kNN) 협업 필터링입니다[2].

전통적으로 kNN의 유사도 행렬은 피어슨 상관관계와 같은 휴리스틱으로 계산되지만 최근 연구[8]에서는 유사도 행렬이 모델 파라미터로 취급되어 작업에 맞게 특별히 학습됩니다. 최근 행렬 인수분해(MF)는 암시적 및 명시적 피드백을 위해 추천 시스템에서 매우 널리 사용되고 있습니다. 초기 연구[13]에서는 특징 행렬을 학습하기 위해 특이값 분해(SVD)가 제안되었습니다. SVD로 학습된 MF 모델은 과적합이 발생하기 쉬운 것으로 나타났습니다.

따라서 정규화된 학습 방법이 제안되었습니다.

항목 예측을 위해 Hu 등[5]과 Pan 등[10]은 대소문자 가중치를 사용한 정규화된 최소제곱 최적화(WR-MF)를 제안합니다. 사례 가중치는 부정적인 사례의 영향을 줄이는 데 사용할 수 있습니다. Hofmann [4]은 항목 추천을 위한 확률론적 잠재 의미 모델을 제안합니다.

슈미트-티엠[14]은 이 문제를 다중 클래스 문제로 변환하고 이진 분류기 세트를 사용하여 해결합니다. 위에서 설명한 모든 항목 예측 작업은 개인화된 순위 데이터 세트에서 평가되지만, 이러한 방법 중 어느 것도 순위를 위해 모델 파라미터를 직접 최적화하지는 않습니다. 대신 사용자가 항목을 선택했는지 여부를 예측하기 위해 최적화합니다.

이번 연구에서는 항목 쌍(즉, 두 항목의 사용자별 순서)을 기반으로 하는 개인화된 순위를 위한 최적화 기준을 도출했습니다. 이 기준에 따라 MF나 적응형 kNN과 같은 최신 모델을 최적화하여 일반적인 학습 방법보다 더 나은 랭킹 품질을 제공할 수 있는 방법을 보여드리겠습니다.

우리의 접근 방식과 Hu 등[5] 및 Pan 등[10]의 WRMF 접근 방식과 최대 마진 행렬 인수분해[15] 사이의 관계에 대한 자세한 논의는 섹션 5에서 확인할 수 있습니다.

4.1.1절에서는 [3]에서와 같이 최적화 기준과 AUC 최적화의 관계에 대해서도 논의할 것입니다.

이 문서에서는 모델 파라미터의 오프라인 학습에 초점을 맞춥니다. 학습 방법을 온라인 학습 시나리오(예: 새로운 사용자가 추가되고 그의 이력이 0에서 1, 2, ... 피드백 이벤트로 증가)로 확장하는 것은 이미 평점 예측이라는 관련 작업을 위해 MF를 위해 연구된 바 있습니다[11]. 동일한   
폴드인 전략을 BPR에도 사용할 수 있습니다. 비협조적 모델로 순위를 매기는 학습에 대한 관련 작업도 있습니다. 한 가지 방향은 순열에 대한 분포를 모델링하는 것입니다 [7, 6]. Burges 등[1]은 경사 하강을 사용하여 순위를 매기기 위한 신경망 모델을 최적화합니다. 이러한 모든 접근 방식은 하나의 순위만 학습합니다. 즉, 개인화되지 않습니다.

이와 달리, 저희 모델은 개인화된 랭킹, 즉 사용자당 하나의 개별 랭킹을 학습하는 협업 모델입니다. 이번 평가에서는 일반적인 추천 설정에서 개인화된 BPR 모델이 개인화되지 않은 랭킹의 이론적 상한선보다 더 우수한 성능을 보인다는 사실을 경험적으로 보여주었습니다.

3. 개인화 랭킹

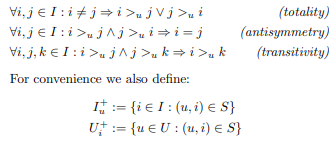
개인화된 순위 작업은 사용자에게 순위가 매겨진 항목 목록을 제공하는 것입니다.

이를 품목 추천이라고도 합니다. 예를 들어 사용자가 구매할 가능성이 있는 품목의 개인화된 순위 목록을 추천하고자 하는 온라인 상점을 들 수 있습니다. 이 백서에서는 사용자의 암묵적 행동(예: 과거 구매)을 통해 순위를 유추해야 하는 시나리오를 살펴봅니다. 암묵적 피드백 시스템의 흥미로운 점은 긍정적인 관찰만 가능하다는 것입니다.

관찰되지 않은 사용자-아이템 쌍(예: 사용자가 아직 아이템을 구매하지 않은 경우)은 실제 부정적인 피드백(사용자가 아이템 구매에 관심이 없음)과 결측치(사용자가 향후 아이템을 구매할 가능성이 있음)가 혼합되어 있습니다.

3.1 Formalization(형식화)

U를 모든 사용자의 집합으로, I를 모든 항목의 집합으로 가정합니다. 이 시나리오에서는 암시적 피드백 S ⊆ U ×I를 사용할 수 있습니다(그림 1의 왼쪽 참조). 이러한 피드백의 예로는 온라인 상점에서의 구매, 동영상 포털에서의 조회수 또는 웹사이트 클릭 등이 있습니다. 이제 추천 시스템의 임무는 사용자에게 모든 항목의 개인화된 총 순위 >u⊂ I^2를 제공하는 것인데, 여기서 >u는 전체 주문의 속성을 충족해야 합니다:



3.2 Analysis of the problem setting( 문제 설정 분석)

앞서 언급했듯이 암시적 피드백 시스템에서는 긍정적인 클래스만 관찰됩니다. 나머지 데이터는 실제로 음수 값과 누락된 값이 혼합되어 있습니다. 결측치 문제에 대처하는 가장 일반적인 접근 방식은 결측치를 모두 무시하는 것이지만, 일반적인 머신러닝 모델은 두 수준을 더 이상 구분할 수 없기 때문에 아무것도 학습할 수 없습니다.

항목 추천자의 일반적인 접근 방식은 다음을 반영하는 항목에 대한 개인화된 점수 ˆxui를 예측하는 것입니다.

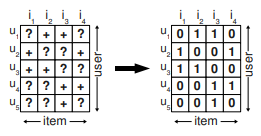
(그림1)

그림 1: 왼쪽에는 관찰된 데이터 S가 표시됩니다. 긍정적인 피드백만 관찰되므로 S에서 직접 학습하는 것은 불가능합니다. 일반적으로 음수 데이터는 행렬을 0 값으로 채워서 생성됩니다.

항목에 대한 사용자의 선호도를 입력합니다. 그런 다음 해당 점수에 따라 항목을 정렬하여 순위를 매깁니다. 항목 추천을 위한 머신 러닝 접근 방식[5, 10]은 일반적으로 (u, i) ∈ S 쌍에 양수 클래스 레이블을 부여하고 (U × I) \ S의 다른 모든 조합에 음수 레이블을 부여하여 S에서 학습 데이터를 생성합니다(그림 1 참조). 그런 다음 이 데이터에 모델을 맞춥니다.

즉, 모델은 S의 요소에 대해서는 1 값을 예측하고 나머지 요소에 대해서는 0 값을 예측하도록 최적화되어 있습니다. 이 접근 방식의 문제점은 모델이 향후에 순위를 매겨야 하는 모든 요소((U × I) \ S)가 학습 알고리즘에 학습 중에 부정적인 피드백으로 제공된다는 것입니다. 즉, 충분한 표현력을 가진 모델(학습 데이터를 정확히 맞출 수 있는)은 0만 예측하기 때문에 전혀 순위를 매길 수 없습니다. 이러한 머신러닝 방법으로 순위를 예측할 수 있는 유일한 이유는 정규화와 같은 과적합을 방지하는 전략 때문입니다.

우리는 항목 쌍을 학습 데이터로 사용하는 다른 접근 방식을 사용하고, 단일 항목에 점수를 매기는 대신 항목 쌍의 순위를 정확하게 매기는 것이 누락된 값을 음수로 대체하는 것보다 문제를 더 잘 나타내기 때문에 최적화합니다. S에서 각 사용자에 대해 >u의 일부를 재구성하려고 합니다.

사용자 u가 항목 i를 본 적이 있다면, 즉 (u, i) ∈ S라면, 사용자가 관찰되지 않은 다른 모든 항목보다 이 항목을 선호한다고 가정합니다. 예를 들어, 그림 2에서 사용자 u1은 항목 i2를 보았지만 항목 i1은 보지 않았으므로 이 사용자는 항목 i1보다 항목 i2를 선호한다고 가정합니다: i2>u i1.

사용자가 두 항목을 모두 본 적이 있는 항목의 경우 선호도를 유추할 수 없습니다. 사용자가 아직 보지 않은 두 개의 항목(예: 사용자 u1의 항목 i1 및 i4)에 대해서도 마찬가지입니다.

이를 공식화하기 위해 학습 데이터 DS 를 생성합니다: U × I × I로 만듭니다: (3page 참고)

(u, i, j) ∈ DS의 의미는 사용자 u가 j보다 i를 선호한다고 가정합니다. >u는 비대칭이므로 음의 경우는 암시적으로 간주됩니다.

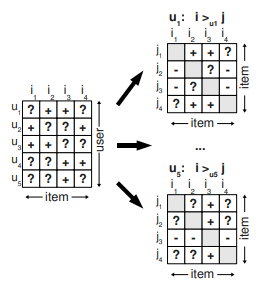
(그림2)

그림 2: 왼쪽에는 관찰된 데이터 S가 표시됩니다. 우리의 접근 방식은 한 쌍의 항목 사이에 사용자별 쌍별 선호도 i>u j를 생성합니다. 오른쪽에서 플러스(+)는 사용자가 항목 j보다 항목 i를 선호함을 나타내고, 마이너스(-)는 i보다 j를 선호함을 나타냅니다.

우리 접근법의 2가지 장점:

1. 학습 데이터는 양수 및 음수 쌍과 누락된 값으로 구성됩니다.

관찰되지 않은 두 항목 사이의 결측값은 정확히 미래에 순위를 매겨야 하는 항목 쌍입니다.

즉, 쌍의 관점에서 볼 때 훈련 데이터 DS와 테스트 데이터는 서로 분리되어 있습니다.

2. 학습 데이터는 실제 순위 지정 목적에 맞게 생성되며, 즉 >u의 관측된 하위 집합 DS가 학습 데이터로 사용됩니다.

4. Bayesian Personalized Ranking

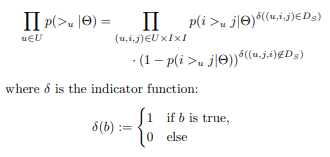
이 섹션에서는 개인화된 랭킹 작업을 해결하기 위한 일반적인 방법을 도출합니다. 이는 개인화된 랭킹을 위한 일반적인 최적화 기준인 BPR-Opt로 구성되며, 이는 p(i >u j|Θ)에 대한 확률 함수와 모델 파라미터 p(Θ)에 대한 사전 확률을 사용하여 문제에 대한 베이지안 분석을 통해 도출됩니다. 이를 순위 통계인 AUC(ROC 곡선 아래 면적)에 비유하여 보여줍니다. BPR-Opt와 관련된 모델 학습을 위해 알고리즘 LearnBPR을 제안합니다. 마지막으로, 두 가지 최신 추천 알고리즘인 행렬 인수분해와 적응형 kNN에 BPROpt와 LearnBPR을 적용하는 방법을 보여드립니다. BPR로 최적화된 이러한 모델은 일반적인 학습 방법보다 더 나은 랭킹을 생성할 수 있습니다.

4.1 BPR Optimization Criterion

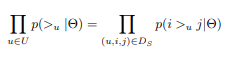
모든 항목 i ∈ I에 대해 올바른 개인화 순위를 찾는 베이지안 공식은 Θ가 임의의 모델 클래스(예: 행렬 인수분해)의 매개변수 벡터를 나타내는 다음의 사후 확률을 최대화하는 것입니다.



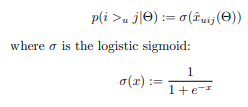
여기서 >u는 사용자 u가 원하는 잠재적 선호도 구조이며, 모든 사용자는 서로 독립적으로 행동하는 것으로 가정합니다. 또한 특정 사용자에 대한 각 항목 쌍(i, j)의 순서는 다른 모든 쌍의 순서와 독립적이라고 가정합니다. 따라서 위의 사용자별 확률 함수 p(>u |Θ)는 먼저 단일 밀도의 곱으로 재작성할 수 있고, 두 번째로 모든 사용자 u ∈ U에 대해 결합할 수 있습니다.



사운드 쌍별 순서 체계의 전체성 및 비대칭성으로 인해 위의 공식은 다음과 같이 단순화할 수 있습니다:



지금까지는 일반적으로 개인화된 총 주문이 보장되지 않습니다. 이를 확립하기 위해서는 이미 언급한 건전한 속성(전체성, 비대칭성 및 전이성)을 충족해야 합니다. 이를 위해 사용자가 실제로 항목 i를 항목 j보다 선호할 개별 확률을 다음과 같이 정의합니다:



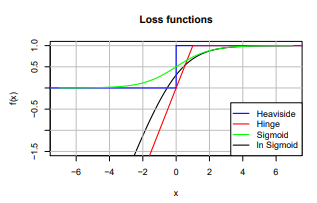


그림 3: AUC 최적화를 위한 손실 함수. 미분할 수 없는 헤비사이드 H(x)는 종종 시그모이드 σ(x)로 근사화됩니다. MLE 유도에서는 대신 ln σ(x)를 사용할 것을 제안합니다.

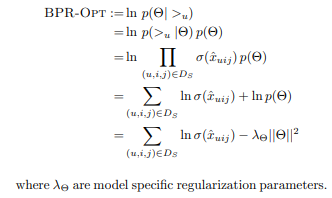
여기서 ˆxuij(Θ)는 모델 파라미터 벡터 Θ의 임의의 실수 값 함수로서 사용자 u, 항목 i, 항목 j 사이의 특수한 관계를 포착합니다.

다시 말해, 일반적인 프레임워크는 u, i, j 사이의 관계를 모델링하는 작업을 행렬 인수분해나 적응형 kNN과 같은 기본 모델 클래스에 위임하여 xˆuij(Θ)의 추정을 담당하게 합니다. 따라서 개인화된 총 주문>u를 통계적으로 모델링하는 것이 가능해집니다. 편의상 다음에서는 ˆxuij 에서 Θ 인수를 생략하겠습니다.

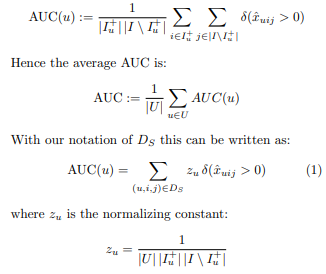
지금까지는 확률 함수에 대해서만 설명했습니다.

개인화된 랭킹 작업의 베이지안 모델링 접근 방식을 완성하기 위해 평균과 분산-공분산 행렬 ΣΘ이 0인 정규 분포인 일반적인 사전 밀도 p(Θ)를 도입합니다.



다음에서는 미지의 하이퍼파라미터의 수를 줄이기 위해 ΣΘ = λΘI로 설정합니다. 이제 최대 사후 추정자를 공식화하여 개인화된 순위 BPR-Opt에 대한 일반적인 최적화 기준을 도출할 수 있습니다. 

4.1 Analogies to AUC optimization

이러한 베이지안 개인화 순위(BPR) 체계의 공식화를 통해 이제 BPR과 AUC의 유사성을 쉽게 파악할 수 있습니다. 사용자당 AUC는 일반적으로 다음과 같이 정의됩니다:

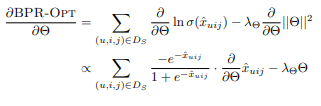
(1)과 BPR-Opt의 유사점은 분명합니다. 정규화 상수 zu를 제외하면 손실 함수만 다를 뿐입니다. AUC는 헤비사이드 함수와 동일한 비분산 손실 δ(x > 0)을 사용합니다:



대신 미분 손실 ln σ(x)를 사용합니다. AUC를 최적화할 때 미분 불가능한 헤비사이드 함수를 대체하는 것이 일반적인 관행입니다[3]. 대체 함수의 선택은 종종 휴리스틱적으로 이루어지며 σ와 유사한 형태의 함수가 사용됩니다(그림 3 참조). 이 백서에서는 MLE에 의해 동기화된 대체 치환 ln σ(x)를 도출했습니다.

4.2 BPR Learning Algorithm

마지막 섹션에서는 개인화된 랭킹을 위한 최적화 기준을 도출했습니다. 이 기준은 차별화할 수 있으므로 경사 하강 기반 알고리즘이 최대화를 위한 당연한 선택입니다. 그러나 앞으로 살펴보겠지만 표준 경사 하강은 우리 문제에 적합한 선택이 아닙니다. 이 문제를 해결하기 위해 학습 삼각형의 부트스트랩 샘플링에 기반한 확률적 경사 하강 알고리즘인 LearnBPR을 제안합니다(그림 4 참조). 우선 모델 파라미터에 대한 BPR-Opt의 기울기는 다음과 같습니다:



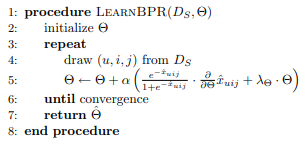
(그림 4)

그림 4: 부트스트래핑 기반 확률적 경사 하강으로 BPR 모델 최적화. 학습률 α 및 정규화 λΘ 사용.

그라디언트 하강을 위한 가장 일반적인 두 가지 알고리즘은 전체 또는 확률적 그라디언트 하강입니다. 첫 번째 경우 각 단계에서 모든 학습 데이터에 대한 전체 경사도를 계산한 다음 학습률 α로 모델 파라미터를 업데이트합니다:



일반적으로 이러한 접근 방식은 '올바른' 방향으로의 하강으로 이어지지만 수렴은 더디게 이루어집니다. DS에는 O(|S| |I|) 훈련 트리플이 있기 때문에 각 업데이트 단계에서 전체 경사도를 계산하는 것은 불가능합니다. 또한 전체 기울기 하강으로 BPR-Opt를 최적화하는 경우에도 훈련 쌍의 왜곡으로 인해 수렴이 제대로 이루어지지 않습니다. 종종 양수인 항목 i를 상상해 보겠습니다. 그러면 많은 사용자 u의 경우 항목 i가 모든 음수 항목 j(지배 클래스)와 비교되기 때문에 손실에 ˆxuij 형식의 항이 많이 있습니다.

따라서 i에 따른 모델 파라미터의 기울기가 대부분 기울기를 지배하게 됩니다. 즉, 매우 작은 학습률을 선택해야 합니다. 둘째, 기울기가 크게 다르기 때문에 정규화가 어렵습니다.

또 다른 인기 있는 접근 방식은 확률적 경사 하강입니다. 이 경우 각 트리플(u, i, j) ∈ DS에 대해 업데이트가 수행됩니다.



일반적으로 이것은 skew 문제에 대한 좋은 접근 방식이지만 훈련 쌍을 탐색하는 순서가 중요합니다. 데이터를 항목별로 또는 사용자별로 탐색하는 일반적인 접근 방식은 동일한 사용자-항목 쌍에 대한 연속적인 업데이트가 너무 많기 때문에, 즉 하나의 사용자-항목 쌍(u, i)에 대해 (u, i, j) ∈ DS를 갖는 많은 j가 있기 때문에 수렴이 제대로 이루어지지 않을 수 있습니다.

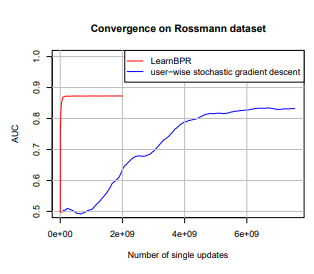
(그림 5)

그림 5: 일반적인 사용자별 확률론적 기울기 하강과 부트스트랩 샘플링을 통한 LearnBPR 알고리즘의 수렴을 경험적으로 비교한 결과입니다.

이 문제를 해결하기 위해 무작위로(균일하게 분포된) 트리플을 선택하는 확률적 경사 하강 알고리즘을 사용하는 것이 좋습니다. 이 접근 방식을 사용하면 연속적인 업데이트 단계에서 동일한 사용자-아이템 조합이 선택될 확률이 적습니다. 어느 단계에서나 중지를 수행할 수 있으므로 교체와 함께 부트스트랩 샘플링 접근 방식을 사용하는 것이 좋습니다. 예제 수가 매우 많고 수렴의 경우 전체 주기의 일부로 충분하기 때문에 데이터를 통한 전체 주기라는 개념을 버리는 것이 특히 유용합니다. 우리는 관찰된 양의 피드백 S의 수에 따라 평가에서 단일 단계의 수를 선형적으로 선택합니다.

어느 단계에서나 중지를 수행할 수 있으므로 교체와 함께 부트스트랩 샘플링 접근 방식을 사용하는 것이 좋습니다. 예제 수가 매우 많고 수렴의 경우 전체 주기의 일부로 충분할 때가 많으므로 데이터를 통한 전체 주기라는 개념을 버리는 것이 특히 유용합니다. 관찰된 긍정적인 피드백의 수에 따라 평가에서 단일 단계의 수를 선형적으로 선택합니다.

그림 5는 일반적인 사용자별 확률론적 그라데이션 하강과 부트스트래핑을 사용한 LearnBPR 접근 방식을 비교1한 것입니다. 모델은 16개의 차원을 가진 BPRMF입니다. 보시다시피 LearnBPR은 사용자별 확률적 경사 하강보다 훨씬 빠르게 수렴합니다.

4.3 Learning models with BPR

아래에서는 아이템 추천을 위한 두 가지 최신 모델 클래스와 제안된 BPR 방법으로 이를 학습하는 방법을 설명합니다. 우리는 행렬 인수분해[5, 12]의 두 가지 다양한 모델 클래스를 선택하고 k-근접 이웃[8]을 학습했습니다.

두 클래스 모두 항목에 대한 사용자의 숨겨진 선호도를 모델링하려고 합니다. 이들의 예측은 사용자-아이템-쌍(u, l)당 실수 ˆxul입니다.

최적화에서는 트리플(u, i, j) ∈ DS가 있으므로 먼저 추정자 ˆxuij를 분해하여 다음과 같이 정의합니다:

이제 ˆxul을 예측하는 모든 표준 협업 필터링 모델을 적용할 수 있습니다. 다른 작업에서와 동일한 모델을 사용하더라도 다른 기준에 따라 최적화한다는 점에 유의하는 것이 중요합니다. 이렇게 하면 순위 작업에 최적화된 기준이 더 나은 순위로 이어집니다. 우리의 기준은 단일 예측 변수 ˆxul을 단일 수치로 회귀시키지 않고 대신 두 예측 변수 ˆxui - xˆuj 의 차이를 분류하려고 시도합니다.

4.3.1 Matrix Factorization

ˆxui를 예측하는 문제는 행렬 X : U × I를 추정하는 작업으로 볼 수 있습니다. 행렬 인수분해를 사용하면 대상 행렬 X는 두 개의 낮은 차원 행렬 W :|U| × k와 H :|I| × k의 행렬 곱으로 근사화됩니다:



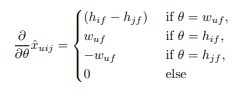
여기서 k는 근사치의 차원/순위입니다. 따라서 예측 공식은 다음과 같이 작성할 수도 있습니다. W의 각 행 wu는 사용자 u를 설명하는 특징 벡터로 볼 수 있으며, 마찬가지로 H의 각 행 hi는 항목 i를 설명합니다:



도트 곱 <-, -> 외에도 일반적으로 [11]과 같이 모든 커널을 사용할 수 있습니다. 행렬 인수분해를 위한 모델 파라미터는 Θ = (W, H)입니다. 모델 파라미터는 잠재 변수로도 볼 수 있으며, 사용자의 관찰되지 않은 취향과 항목의 관찰되지 않은 속성을 모델링합니다.

일반적으로 최소 제곱에 대한 Xˆ에서 X에 대한 최상의 근사치는 특이값 분해(SVD)를 통해 얻을 수 있습니다. 머신 러닝 작업의 경우 SVD가 과도하게 적합하다고 알려져 있으며, 따라서 정규화된 최소제곱 최적화, 비음수 인수분해, 최대 마진 인수분해 등 다른 많은 행렬 인수분해 방법이 제안되었습니다.

순위를 매기는 작업, 즉 사용자가 한 항목을 다른 항목보다 선호하는지 여부를 추정하는 작업의 경우, 더 나은 접근 방식은 BPR-Opt 기준에 따라 최적화하는 것입니다. 이는 제안된 알고리즘인 LearnBPR을 사용하여 달성할 수 있습니다. 앞서 LearnBPR로 최적화할 때 설명한 것처럼 모든 모델 파라미터 θ에 대한 ˆxuij의 기울기만 알면 됩니다.

행렬 인수분해 모델의 경우 도함수는 다음과 같습니다: 

또한 세 가지 정규화 상수를 사용합니다. 사용자 특징 W에는 λW 하나, 항목 특징 H에는 두 가지 정규화 상수, 즉 hif의 포지티브 업데이트에 사용되는 λH+와 hjf의 네거티브 업데이트에 사용되는 λH-가 있습니다.

4.3.2 Adaptive k- Nearset - Neighbor

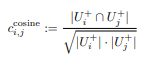
가장 가까운 이웃 방법은 협업 필터링에서 매우 널리 사용됩니다. 이 방법은 항목(항목 기반) 또는 사용자(사용자 기반) 간의 유사성 측정값에 의존합니다.

아래에서는 항목 기반 방법이 일반적으로 더 나은 결과를 제공하므로 항목 기반 방법을 설명하지만 사용자 기반 방법도 유사하게 작동합니다. 사용자 u와 항목 i에 대한 예측은 사용자가 과거에 본 다른 모든 항목, 즉 Iu+에 대한 i의 유사성에 따라 달라진다는 개념입니다. 종종 Iu+와 가장 유사한 k 개의 항목, 즉 가장 가까운 이웃 항목만 고려됩니다. 항목 간의 유사성을 신중하게 선택하면 Iu+의 모든 항목과 비교할 수도 있습니다. 항목 예측의 경우 항목 기반 k-최근접 이웃 모델이 사용됩니다:



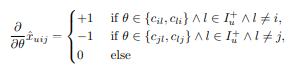
여기서 C : I × I는 대칭적인 항목 상관관계/항목 유사도 행렬입니다. 따라서 kNN의 모델 파라미터는 Θ = C입니다.

C를 선택하는 일반적인 방법은 코사인 벡터 유사도와 같은 휴리스틱 유사도 측정값을 적용하는 것입니다:



더 나은 전략은 유사도 측정값 C를 학습하여 문제에 적용하는 것입니다. 이는 C를 모델 파라미터로 직접 사용하거나 항목 수가 너무 많은 경우 H : I × k로 C의 인수분해 HHt를 학습할 수 있습니다. 다음 및 평가에서는 C를 인수분해하지 않고 직접 학습하는 첫 번째 접근 방식을 사용합니다.

순위를 매기기 위한 kNN 모델 최적화를 위해 다시 한 번 BPR 최적화 기준을 적용하고 LearnBPR 알고리즘을 사용합니다. 알고리즘을 적용하기 위해 모델 파라미터 C에 대한 ˆxuij의 기울기는 다음과 같습니다:



두 개의 정규화 상수, 즉 cil 업데이트의 경우 λ+, cjl 업데이트의 경우 λ-가 있습니다.

5. Relations to other methods(다른 방법과의 관계)

제안된 순위 지정 방법과 두 가지 추가 항목 예측 모델의 관계에 대해 설명합니다.

5.1 Weighted Regularized Matrix Factorization (WR-MF)

(가중 정규화 행렬 인수분해)

Pan 등[10]과 Hu 등[5]은 암시적 피드백으로부터 항목 예측을 위한 행렬 인수분해 방법을 제시했습니다. 따라서 모델 클래스는 4.3.1절에서 설명한 것과 동일합니다. 즉, 행렬 W :|U| × k 및 H :|U| × k가 있는 Xˆ := W Ht입니다.

최적화 기준과 학습 방법은 우리의 접근 방식과 크게 다릅니다. 그들의 방법은 제곱 손실을 최소화하는 SVD를 채택한 것입니다. 또한 과적합을 방지하기 위한 정규화와 오차 함수에 가중치를 부여하여 긍정적인 피드백의 영향을 증가시킵니다. 총 최적화 기준은 다음과 같습니다:



여기서 cui는 모델 파라미터가 아니라 각 튜플(u, i)에 대해 선험적으로 주어진 가중치입니다. Hu 등은 긍정적 피드백에 대한 cui를 추정하기 위한 추가 데이터를 가지고 있으며, 나머지는 cui = 1로 설정했습니다. Pan 등은 긍정적 피드백에 대해 cui = 1을 설정하고 나머지는 더 낮은 상수를 선택할 것을 제안합니다.

우선, 이 최적화는 BPR로서 페어 수준(두 항목)이 아닌 인스턴스 수준(한 항목)에서 이루어지고 있다는 점이 분명합니다. 이 외에도 이 최적화는 정규 분포 확률 변수에 대한 최소제곱에 해당하는 것으로 알려진 최소자승법입니다. 그러나 항목 예측 작업은 실제로 회귀(정량적)가 아니라 분류(정성적) 작업이므로 로지스틱 최적화가 더 적절합니다.

WR-MF의 강점은 양이 아닌 쌍에 대해 cui가 일정하다면 O(iter (|S| k^2 +k^3 (|I|+|U|)))로 학습할 수 있다는 점입니다. 우리의 평가에 따르면 학습해야 할 트리플이 훨씬 더 많음에도 불구하고 LearnBPR은 일반적으로 m \* |S| 단일 업데이트 단계의 하위 샘플 후에 수렴합니다.

5.2 Maximum Margin Matrix Factorization (MMMF)

(최대 마진 행렬 인수분해)

Weimer 등[15]은 서수 순위를 매길 때 최대 마진 행렬 인수분해 방법(MMMF)을 사용합니다. 이들의 MMMF는 평가에 대한 명시적인 피드백이 있는 시나리오를 위해 설계되었습니다.

이 순위 MMMF는 암시적 피드백 데이터 세트에는 적합하지 않지만, 관찰되지 않은 모든 항목에 '평점'을 0으로, 관찰된 항목에 1을 부여하여 시나리오에 적용할 수 있습니다(그림 1 참조). 이렇게 수정하면 최소화해야 할 최적화 기준이 행렬 인수분해에 적용되는 BPR과 매우 유사해집니다:



한 가지 차이점은 오류 함수가 다르다는 점입니다. 저희의 hinge 손실은 매끄럽고 MLE에 의해 동기화됩니다. 또한, 우리의 BPR-Opt 기준은 일반적이며 여러 모델에 적용할 수 있는 반면, 그들의 방법은 MF에 특화되어 있습니다.

이 외에도 MMMF에 대한 학습 방법은 일반적인 접근 방식인 LearnBPR과 다릅니다. 이 학습 방법은 희박한 명시적 데이터, 즉 누락된 값이 많다고 가정하여 암시적 설정보다 훨씬 적은 수의 쌍이 있다고 가정하여 작동하도록 설계되었습니다. 그러나 이 학습 방법을 암시적 피드백 데이터 세트에 적용하면 위에서 설명한 것처럼 데이터가 밀집되어야 하고 훈련 쌍 DS의 수는 O(|S| |I|)가 됩니다. 우리의 방법인 LearnBPR은 DS에서 부트스트랩하여 이러한 상황을 처리할 수 있습니다(4.2절 참조).

6. 평가

평가에서는 BPR을 통한 학습과 다른 학습 접근 방식을 비교합니다. 널리 사용되는 두 가지 모델 클래스인 행렬 인수분해(MF)와 k-근접 이웃(kNN)을 선택했습니다. MF 모델은 공동 평가 예측이라는 관련 작업에서 베이지안 모델 URP [9] 및 PLSA [4]를 포함한 다른 많은 모델보다 우수한 성능을 보이는 것으로 알려져 있습니다[12]. 본 평가에서 행렬 인수분해 모델은 세 가지 다른 방법, 즉 SVD-MF, WR-MF [5, 10], 그리고 우리의 BPR-MF로 학습됩니다. kNN의 경우, 코사인 벡터 유사도(코사인-kNN)를 BPR 방법(BPR-kNN)을 사용해 최적화한 모델과 비교합니다.

또한, 사용자에 따라 각 항목에 가중치를 부여하는 기준선인 가장 인기 있는 항목에 대한 결과를 보고합니다(예: ).

또한, 개인화되지 않은 모든 랭킹 방식에 대한 이론적 상한선(npmax)을 제공합니다.

6.1 Datasets

서로 다른 두 애플리케이션의 두 데이터 세트를 사용합니다.

Rossmann 데이터 세트는 온라인 상점의 데이터 세트입니다. 여기에는 4000개 품목에 대한 10, 000명의 사용자 구매 내역이 포함되어 있습니다. 총 426, 612건의 구매가 기록되어 있습니다. 이 작업은 사용자가 다음에 구매하고자 하는 품목의 개인화된 목록을 예측하는 것입니다.

두 번째 데이터 세트는 넷플릭스의 DVD 대여 데이터 세트입니다. 이 데이터 세트에는 사용자가 일부 영화에 대해 별 1~5개의 명시적인 평점을 제공하는 사용자의 평점 행동이 포함되어 있습니다. 암시적 피드백 작업을 해결하고자 하므로 데이터 세트에서 평점 점수를 제거했습니다.

이제 사용자가 영화에 별점을 매길 가능성이 있는지 예측하는 것이 과제입니다. 다시 한 번 평점을 받을 가능성이 가장 높은 영화부터 시작하여 개인화된 순위 목록이 필요합니다. 넷플릭스의 경우 사용자 10, 000명, 565, 738개의 평점 액션이 포함된 5000개의 항목으로 구성된 하위 샘플을 만들었습니다. 모든 사용자가 최소 10개 이상의 항목(10)을 가지고 있고 각 항목에 최소 10명의 사용자가 있도록 하위 표본을 그립니다: 10

6.2 Evaluation Methodology

각 사용자의 기록에서 무작위로 하나의 액션(하나의 유저아이템 쌍)을 제거하는, 즉 사용자 u당 Iu+에서 하나의 항목을 제거하는 리브 원 아웃 평가 방식을 사용합니다. 이렇게 하면 분리된 훈련 세트 S train과 테스트 세트 S test가 생성됩니다. 그런 다음 모델은 Strain에서 학습되고 예측된 개인화 순위는 테스트 세트 S test에서 평균 AUC 통계에 의해 평가됩니다:



여기서 사용자 U당 평가 쌍은 다음과 같습니다:



AUC 값이 높을수록 더 좋은 품질을 나타냅니다. 무작위 추측 방법의 사소한 AUC는 0.5이고 달성 가능한 최고 품질은 1입니다.

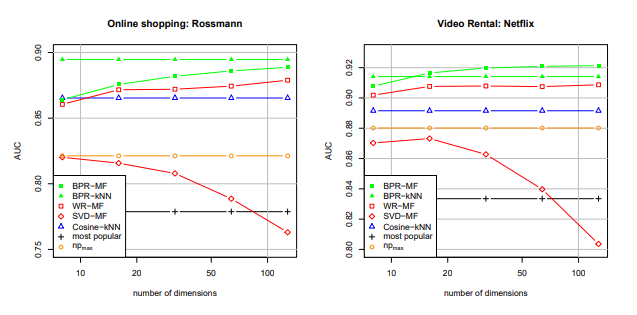
각 라운드에서 새로운 기차/테스트 분할을 도출하여 모든 실험을 10회 반복했습니다. 모든 방법의 하이퍼파라미터는 첫 번째 라운드에서 그리드 검색을 통해 최적화되고 이후 나머지 9번의 반복에서 일정하게 유지됩니다.

6.3 Results and Discussion

그림 6은 두 데이터 세트의 모든 모델에 대한 AUC 품질을 보여줍니다. 우선, 두 가지 BPR 최적화 방법이 예측 품질에서 다른 모든 방법보다 우수한 성능을 보이는 것을 알 수 있습니다. 동일한 모델을 서로 비교하면 최적화 방법의 중요성을 알 수 있습니다. 예를 들어 모든 MF 방법(SVD-MF, WR-MF 및 BPR-MF)은 정확히 동일한 모델을 공유하지만 예측 품질은 크게 다릅니다.

SVD-MF는 요소별 최소 제곱과 관련하여 학습 데이터에 가장 잘 맞는 것으로 알려져 있지만, 과적합이 발생하기 때문에 머신 러닝 작업에는 적합하지 않은 예측 방법입니다. 이는 차원 수가 증가함에 따라 SVD-MF의 품질이 떨어지기 때문이라고 볼 수 있습니다. WR-MF는 랭킹 작업에 더 성공적인 학습 방법입니다.

정규화로 인해 성능이 떨어지지 않고 차원 수가 증가함에 따라 꾸준히 상승합니다. 그러나 두 데이터 세트 모두에서 순위를 매기는 작업에서는 BPR-MF가 WR-MF보다 성능이 월등히 뛰어납니다.



(그림6)

그림 6: 로스만 데이터 세트와 넷플릭스 하위 샘플에 대한 ROC 곡선 아래 면적(AUC) 예측 품질.

가중 정규화 행렬 인수분해(WR-MF)[5, 10], 특이값 분해(SVD-MF), k-최근접 이웃(코사인-kNN)[2] 및 가장 많이 사용되는 모델과 비교하여 행렬 인수분해 BPR-MF와 k-최근접 이웃 BPR-kNN에 대한 BPR 최적화를 비교했습니다. 인수분해 방법인 BPR-MF, WR-MF 및 SVD-MF의 경우 모델 차원이 8차원에서 128차원으로 증가합니다. 마지막으로 npmax는 개인화되지 않은 모든 랭킹 방법의 이론적 상한선입니다.

예를 들어 넷플릭스에서 BPR-MF로 최적화된 8개의 차원을 가진 MF 모델은 WR-MF로 최적화된 128개의 차원을 가진 MF 모델과 비슷한 품질을 달성합니다.

요약하자면, 이번 결과는 올바른 기준에 따라 모델 파라미터를 최적화하는 것이 얼마나 중요한지 보여줍니다. 그리고 경험적 결과에 따르면 LearnBPR에서 학습한 BPR-Opt 기준은 암시적 피드백을 통해 개인화된 순위를 매기는 다른 최신 방법보다 성능이 뛰어납니다. 이러한 결과는 문제 분석(섹션 3.2)과 MLE에서 BPR-Opt의 이론적 도출에 의해 정당화됩니다.

6.4 Non-personalized ranking

마지막으로, 개인화된 랭킹 방법의 AUC 품질을 가능한 최상의 비개인화 랭킹 방법과 비교합니다. 개인화된 랭킹 방법과 달리, 비개인화된 랭킹 방법은 모든 사용자에 대해 동일한 랭킹을 생성합니다. 테스트 세트 S-test 2에서 순위를 최적화하여 모든 비개인화 랭킹 방법에 대한 이론적 상한 np-max를 계산합니다. 그림 6은 CosinekNN과 같은 단순한 개인화 방법도 상한 np-max를 능가하며, 따라서 모든 비개인화 방법도 대체로 능가한다는 것을 보여줍니다.

7. 결론

이 문서에서는 개인화된 랭킹을 위한 일반적인 최적화 기준과 학습 알고리즘을 제시했습니다.  
최적화 기준인 BPR-Opt는 문제에 대한 베이지안 분석에서 도출된 최대 사후 추정치입니다.  
BPR-Opt와 관련된 학습 모델의 경우, 부트스트랩 샘플링을 통한 확률적 경사 하강을 기반으로 하는 일반 학습 알고리즘인 LearnBPR을 제시했습니다. 이 일반적인 방법을 행렬 인수분해와 적응형 kNN이라는 두 가지 최신 추천 모델에 어떻게 적용할 수 있는지 시연했습니다. 평가 결과, 개인화된 랭킹 작업의 경우 BPR로 학습한 모델이 다른 기준에 따라 최적화된 동일한 모델보다 성능이 뛰어나다는 것을 경험적으로 보여주었습니다. 우리의 결과는 예측 품질이 모델뿐만 아니라 최적화 기준에 따라 크게 달라진다는 것을 보여줍니다. 이론적 및 실증적 결과는 모두 BPR 최적화 방법이 개인화된 랭킹이라는 중요한 작업에 적합한 선택임을 나타냅니다.