

인공지능 개론

L07 Latent Factor Model

국민대학교
소프트웨어융합대학원
박하명

추천시스템

내가 이 드라마를 본다면, 별점을 몇점을 줄까?
내 예상별점이 높은 드라마를 추천해줘..!



한국 TV 인기 순위 1위

WATCHA

부부의 세계

2020 • JTBC • 스릴러/드라마/TV드라마

평점 ★4.5 (4,294명) • 예상 ★4.3



박하명 님의 취향저격 베스트 콘텐츠

NETFLIX

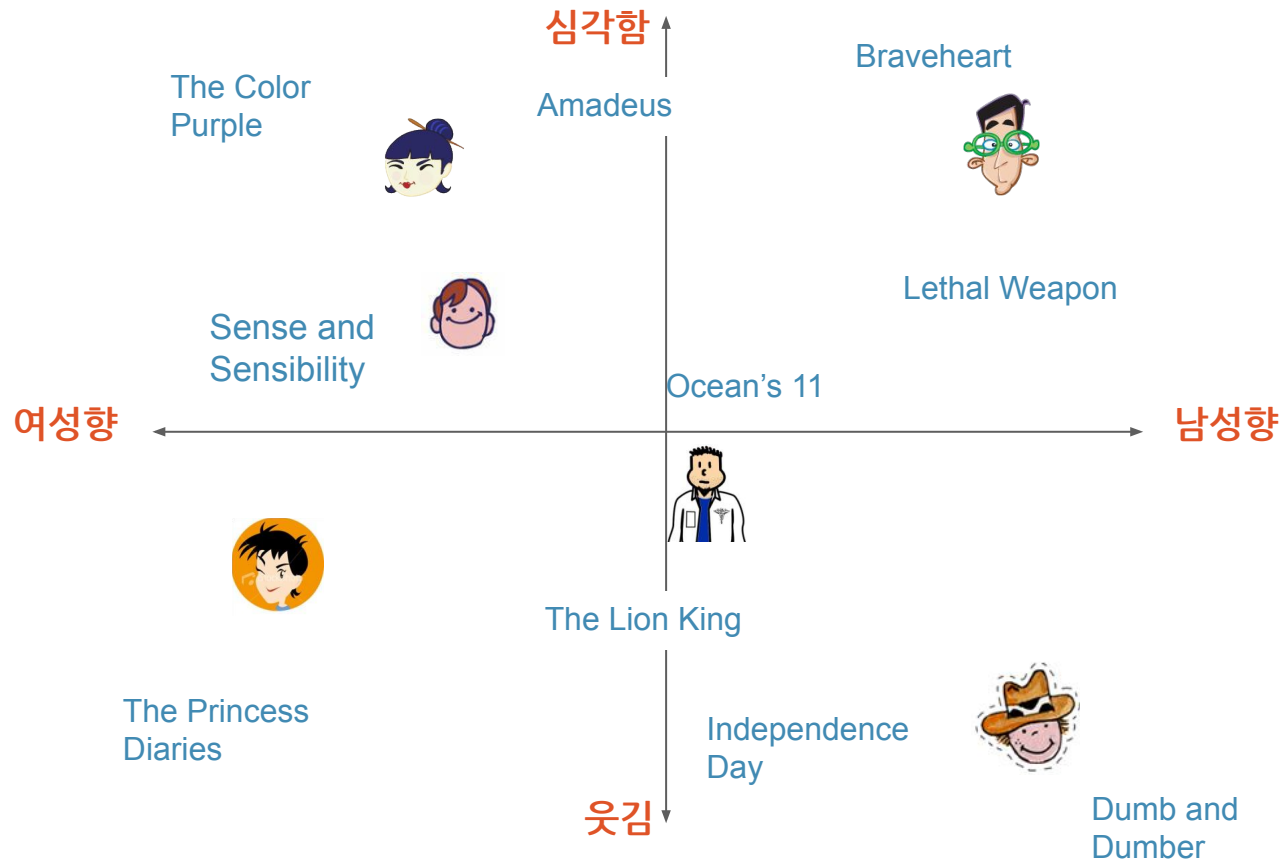


별점 예측 방법

- Collaborative Filtering
 - Item-Item Collaborative Filtering
 - User-User Collaborative Filtering
- **Latent Factor Model**

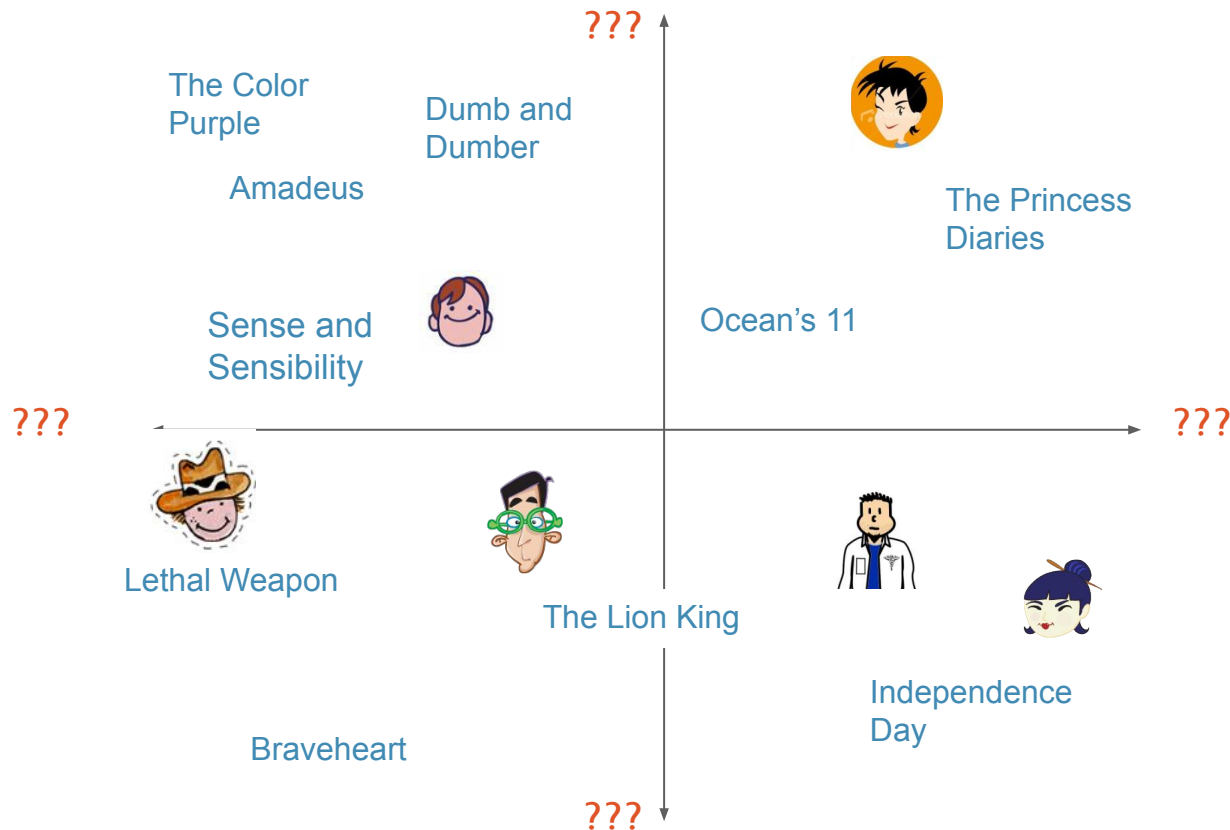
Factor Model

- 사용자와 아이템을 **요소(Factor)**들로 나타낼 수 있다고 보는 모델





















Latent Factor Model

- 사용자와 아이템을 **잠재적인 요소(Latent Factor)**들로 나타낼 수 있다고 보는 모델
= 뭔진 모르겠지만 아무튼 어떤 요소



Matrix Factorization

- 사용자와 영화를 같은 차원의 공간에 매핑한다.
= 각각의 사용자와 영화를 같은 차원의 벡터로 표현한다.

	 A	 B	 C	 D	 E	 F	 G	 H	 I	 J	 K	 L
a  터널		4		5			5			3		1
b  SKY캐슬	3	1	2			4			4	5		
c  보이스		5	3	4		3		2	1		4	2
d  식객		2			4			5		4	2	
e  이태원 클라쓰	5	2					2	4	3	4		
f  치즈캐슬		4			2			3		3		1

Matrix Factorization

- 각각의 사용자와 영화를 같은 차원의 벡터로 표현한다.

Item Matrix P			User Matrix Q^T											
0.3	0.4	0.4		4		5		5	?		3		1	
0.4	0.4	0.1	3	1	2			4			4	5		
0.4	0.2	0.8		5	3	4		3		2	1		4	2
0.7	0.5	0.9		2			4			5		4	2	
0.1	0.1	0.6	5	2					2	4	3	4		
0.1	0.3	0.1		4			2			3		3		1

Matrix Factorization

- 사용자와 영화가 벡터로 표현됐다면, 평점을 예측할 수 있다.

Item Matrix P			User Matrix Q^T											
0.3	0.4	0.4	0.4	0.1	0.4	0.1	0.4	0.2	0.4	0.1	0.4	0.1		
0.4	0.4	0.1	0.3	0.3	0.5	0.6	0.3	0.6	0.1	0.2	0.3	0.4	0.3	0.2
0.4	0.2	0.8	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.7	0.3	0.3	0.3	0.7
0.7	0.5	0.9												
0.1	0.1	0.6	4		5			5	4.2		3		1	
0.1	0.3	0.1	3	1	2			4			4	5		
			5	3	4		3		2	1		4	2	
			2			4			5		4	2		
			5	2				2	4	3	4			
				4		2			3		3		1	

Matrix Factorization

- 사용자 매트릭스와 아이템 매트릭스를 어떻게 구할 수 있을까?
⇒ 예측 결과가 실제 값과 비슷해지도록 최적화!



	A	B	C	D	E	F	G	H	I	J	K	L
a	4		5			5			3		1	
b	3	1	2			4			4	5		
c		5	3	4		3		2	1		4	2
d		2			4			5		4	2	
e	5	2					2	4	3	4		
f		4			2			3		3		1

$$\arg \min_{P, Q} \sum_{(i, x) \in R} (r_{xi} - p_i \cdot q_x)^2$$

$$\begin{matrix}
 \approx \\
 \begin{matrix}
 0.3 & 0.4 & 0.4 \\
 0.4 & 0.4 & 0.1 \\
 0.4 & 0.2 & 0.8 \\
 0.7 & 0.5 & 0.9 \\
 0.1 & 0.1 & 0.6 \\
 0.1 & 0.3 & 0.1
 \end{matrix}
 \end{matrix}
 \times
 \begin{matrix}
 0.4 & 0.1 & 0.4 & 0.1 & 0.4 & 0.1 & 0.4 & 0.2 & 0.4 & 0.1 & 0.4 & 0.1 \\
 0.3 & 0.3 & 0.5 & 0.6 & 0.3 & 0.6 & 0.1 & 0.2 & 0.3 & 0.4 & 0.3 & 0.2 \\
 0.3 & 0.7 & 0.3 & 0.7 & 0.3 & 0.7 & 0.3 & 0.7 & 0.3 & 0.3 & 0.3 & 0.7
 \end{matrix}$$

가설, 비용, 업데이트

- 가설함수:

$$H(i, x) = p_i \cdot q_x$$

- 비용:

$$cost(P, Q) = \sum_{(i,x) \in R} (r_{xi} - H(i, x))^2$$

- 업데이트:

$$P = P - \alpha \frac{\partial cost(P, Q)}{\partial P}$$

$$Q = Q - \alpha \frac{\partial cost(P, Q)}{\partial Q}$$

가설, 비용, 업데이트



















- 비용:

$$cost(P, Q) = \sum_{(i,x) \in R} (r_{xi} - H(i, x))^2$$

$$\propto \frac{1}{|R|} \sum_{(i,x) \in R} (r_{xi} - H(i, x))^2 = MSE$$

Training, Test

- 잘 학습되었는지를 검증하려면?
 - 학습 데이터와 검증 데이터를 분리

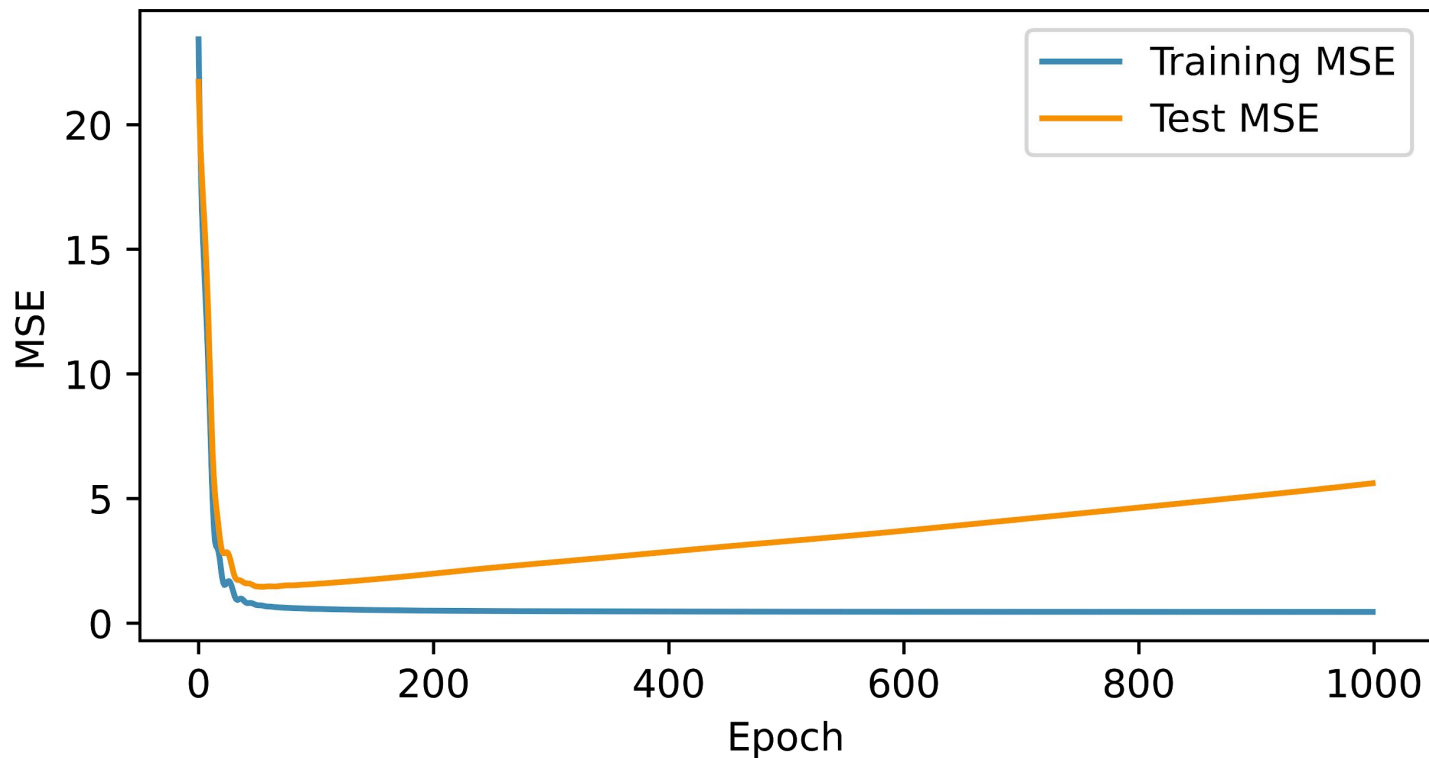
	 A	 B	 C	 D	 E	 F	 G	 H	 I	 J	 K	 L
a  터널		4		5			5			3		1
b  SKY캐슬	3	1	2			4			4	5		
c  보이스		5	3	4		3		2	1		4	2
d  삼사를		2			4			5		4	2	
e  이태원 클라쓰	5	2					2	4	3	4		
f  시크릿		4			2			3		3		1

Training Set

Test Set

Overfitting

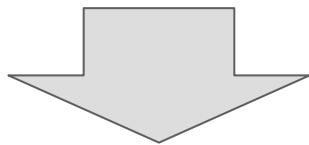
- 실제로 학습해보면, 학습이 잘 되는가 싶은데...
 - 학습이 진행될수록 MSE 값이 점점 증가한다...? → **Overfitting 발생**



Regularization

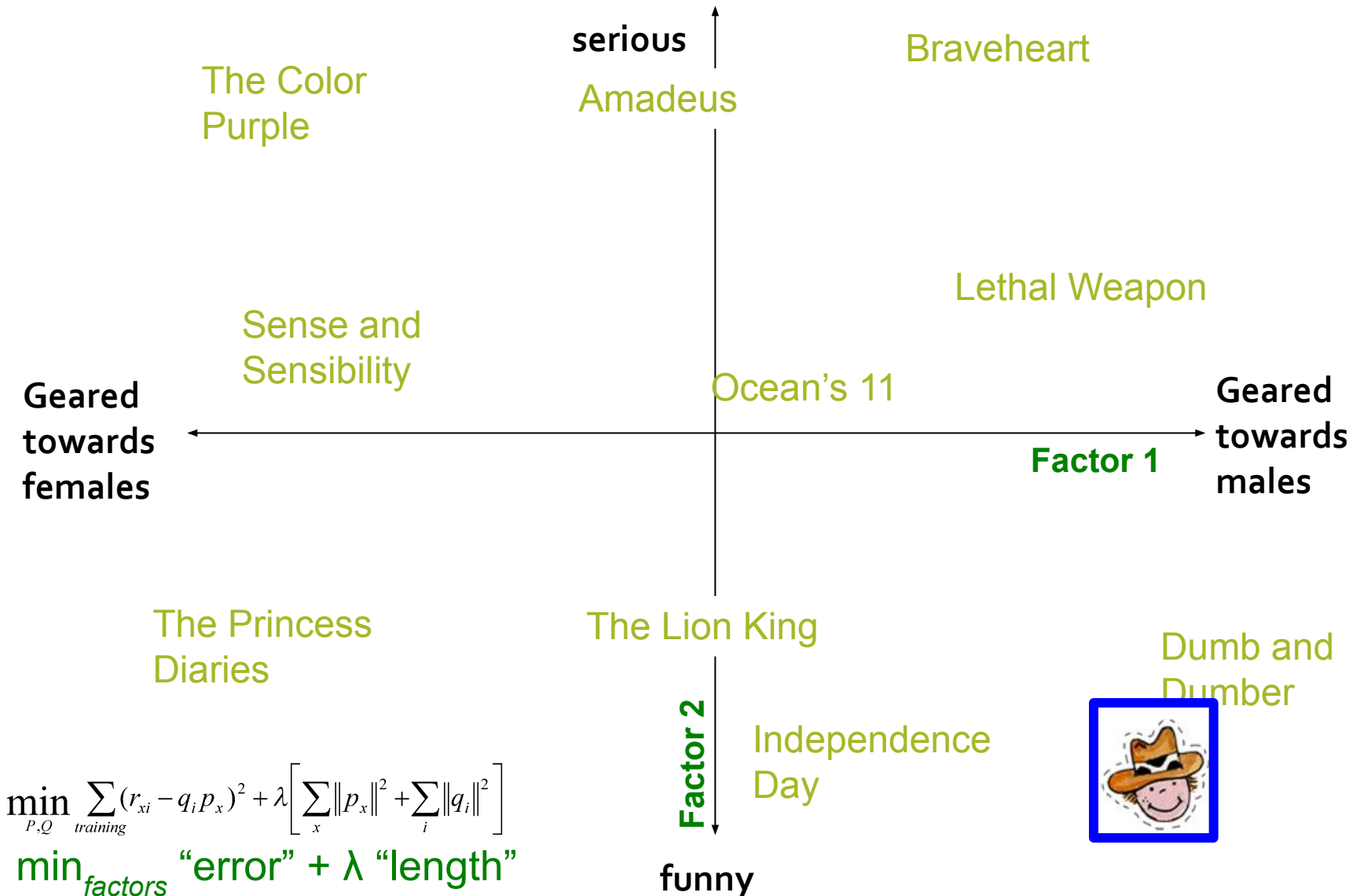
- Training Data에 너무 매몰되지 않도록, latent vector가 너무 큰 값을 갖는 (범위를 벗어나는) 경우 패널티를 주자 → **정규화 (Regularization)**

$$cost(P, Q) = \sum_{(i,x) \in R} (r_{xi} - H(i, x))^2$$

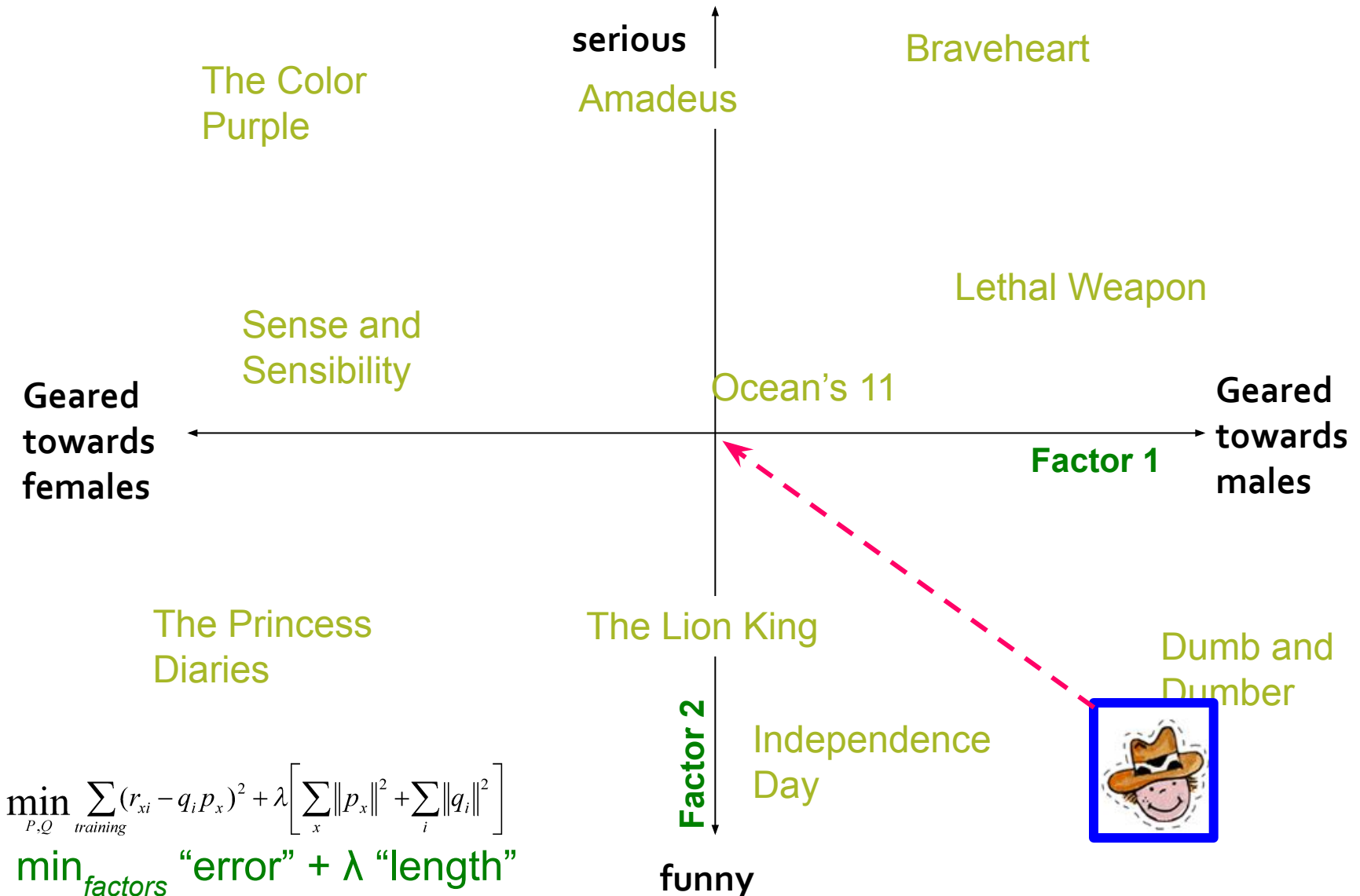


$$cost(P, Q) = \sum_{(i,x) \in R} (r_{xi} - H(i, x))^2 + \lambda_1 \sum_i ||p_i||_2^2 + \lambda_2 \sum_x ||q_x||_2^2$$

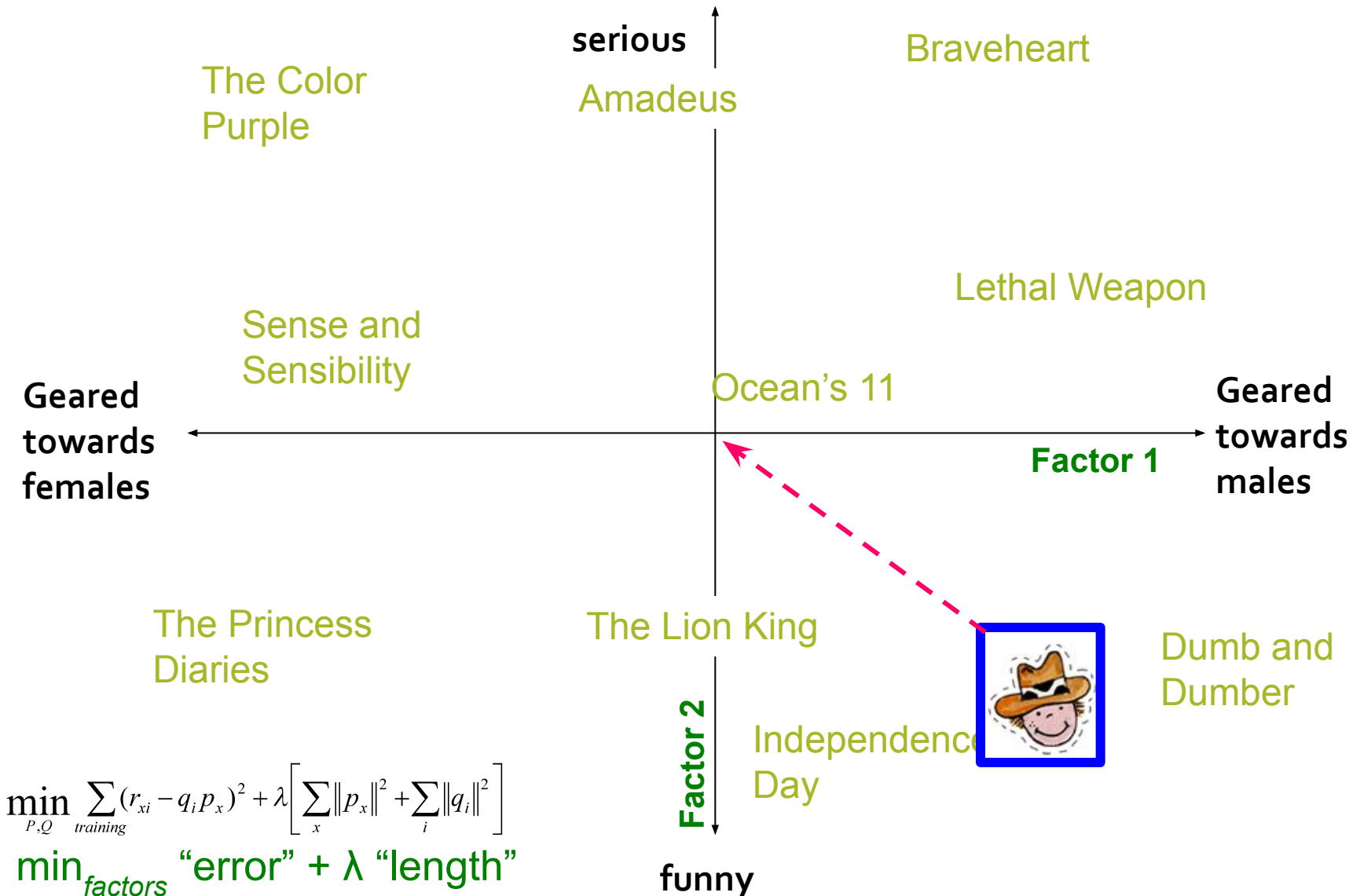
The Effect of Regularization



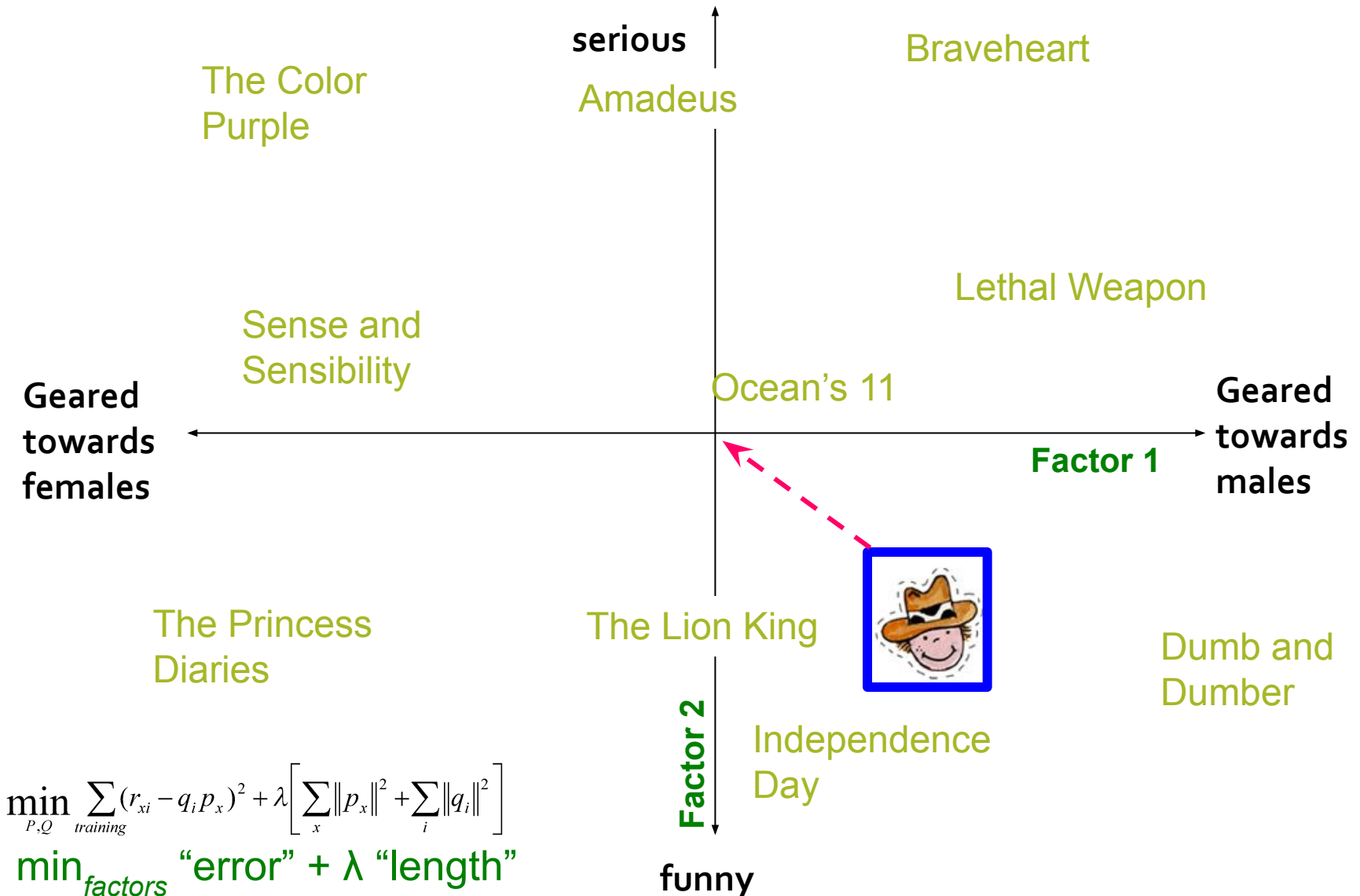
The Effect of Regularization



The Effect of Regularization



The Effect of Regularization



Global Baseline Estimate (다시보기)

이미 높은 평점을 받은 드라마에는 나도 높은 평점을 주지 않을까?

진원이는 깐깐한 편인데, 평균보다 조금 낮게 평점을 주지 않을까?

- 진원이가 드라마 "이태원 클라쓰"를 보고 매길 평점 예측하기
 - 문제: 진원이는 "이태원 클라쓰"와 비슷한 드라마를 본 적이 없다...!
- 평점 가늠해보기 (Global Baseline Estimate)
 - 평균 드라마 평점: 3.7점
 - "이태원 클라쓰"의 평점 평균: 4.2점 (평균보다 0.5점 높음)
 - 진원이의 평점 평균: 3.5점 (평균보다 0.2점 낮음)
 - 기본 점수 (Global baseline) 예측: $3.7 + 0.5 - 0.2 = 4.0$ 점

Latent Factor Model + Bias

$$H(i, x) = \mu + b_i + b_x + p_i \cdot q_x$$

Overall
mean rating

Bias for
movie i

Bias for
user x

User-Movie
interaction

$$\begin{aligned} \text{cost}(P, Q) = & \sum_{(x,i) \in R} (r_{xi} - H(i, x))^2 \\ & + \left(\lambda_1 \sum_i \|p_i\|_2^2 + \lambda_2 \sum_x \|q_x\|_2^2 + \lambda_3 \sum_i \|b_i\|_2^2 + \lambda_4 \sum_x \|b_x\|_2^2 \right) \end{aligned}$$

Question?