Factorization Models for Recommender Systems and Other Applications

Part I

Lars Schmidt-Thieme, Steffen Rendle

Information Systems and Machine Learning Lab (ISMLL)
Institute of Computer Science
University of Hildesheim, Germany

Institute of Computer Science,
University of Konstanz, Germany

KDD 2012 — Tutorial
August 12, 2012, Beijing
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
### Factorization Models for Recommender Systems and Other Applications

1. Matrix Factorization Models

#### PCA

<table>
<thead>
<tr>
<th>V1</th>
<th>V2</th>
<th>V3</th>
<th>V4</th>
<th>V5</th>
<th>V6</th>
<th>V7</th>
<th>V8</th>
<th>V9</th>
<th>V10</th>
<th>V11</th>
<th>V12</th>
<th>V13</th>
<th>V14</th>
<th>V15</th>
<th>V16</th>
<th>V17</th>
<th>V18</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>17</td>
<td>13</td>
<td>14</td>
<td>16</td>
<td>4</td>
<td>9</td>
<td>14</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>11</td>
<td>13</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>6</td>
<td>5</td>
<td>12</td>
<td>1</td>
<td>9</td>
<td>1</td>
<td>7</td>
<td>12</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>13</td>
<td>18</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>8</td>
<td>6</td>
<td>14</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>9</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>18</td>
<td>16</td>
<td>15</td>
<td>19</td>
<td>8</td>
<td>6</td>
<td>11</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>13</td>
<td>13</td>
<td>22</td>
<td>8</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td>10</td>
<td>3</td>
<td>8</td>
<td>11</td>
<td>7</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>13</td>
<td>16</td>
<td>21</td>
<td>9</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td>10</td>
<td>4</td>
<td>7</td>
<td>12</td>
<td>7</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>11</td>
</tr>
<tr>
<td>17</td>
<td>22</td>
<td>19</td>
<td>10</td>
<td>16</td>
<td>10</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td>12</td>
<td>2</td>
<td>8</td>
<td>11</td>
<td>6</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>18</td>
<td>20</td>
<td>10</td>
<td>24</td>
<td>10</td>
<td>6</td>
<td>9</td>
<td>21</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>7</td>
<td>18</td>
<td>4</td>
<td>9</td>
<td>1</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>21</td>
<td>12</td>
<td>15</td>
<td>9</td>
<td>7</td>
<td>11</td>
<td>1</td>
<td>9</td>
<td>3</td>
<td>8</td>
<td>14</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>21</td>
<td>12</td>
<td>15</td>
<td>9</td>
<td>7</td>
<td>11</td>
<td>1</td>
<td>9</td>
<td>3</td>
<td>9</td>
<td>14</td>
<td>5</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>21</td>
<td>18</td>
<td>11</td>
<td>24</td>
<td>10</td>
<td>9</td>
<td>6</td>
<td>19</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>7</td>
<td>14</td>
<td>4</td>
<td>11</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>22</td>
<td>20</td>
<td>12</td>
<td>23</td>
<td>10</td>
<td>8</td>
<td>5</td>
<td>20</td>
<td>0</td>
<td>13</td>
<td>0</td>
<td>6</td>
<td>13</td>
<td>4</td>
<td>11</td>
<td>0</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>23</td>
<td>18</td>
<td>19</td>
<td>18</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>12</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>10</td>
<td>15</td>
<td>8</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>24</td>
<td>17</td>
<td>16</td>
<td>17</td>
<td>15</td>
<td>8</td>
<td>6</td>
<td>14</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>9</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>17</td>
<td>14</td>
<td>16</td>
<td>8</td>
<td>6</td>
<td>14</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>26</td>
<td>19</td>
<td>17</td>
<td>15</td>
<td>16</td>
<td>8</td>
<td>5</td>
<td>14</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>27</td>
<td>19</td>
<td>15</td>
<td>16</td>
<td>16</td>
<td>8</td>
<td>6</td>
<td>14</td>
<td>1</td>
<td>15</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>28</td>
<td>17</td>
<td>17</td>
<td>16</td>
<td>19</td>
<td>8</td>
<td>6</td>
<td>11</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>29</td>
<td>18</td>
<td>17</td>
<td>15</td>
<td>19</td>
<td>8</td>
<td>6</td>
<td>11</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>30</td>
<td>22</td>
<td>19</td>
<td>10</td>
<td>16</td>
<td>10</td>
<td>5</td>
<td>17</td>
<td>1</td>
<td>12</td>
<td>2</td>
<td>8</td>
<td>11</td>
<td>6</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>31</td>
<td>18</td>
<td>20</td>
<td>10</td>
<td>18</td>
<td>6</td>
<td>8</td>
<td>17</td>
<td>1</td>
<td>14</td>
<td>1</td>
<td>5</td>
<td>16</td>
<td>4</td>
<td>7</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>32</td>
<td>18</td>
<td>16</td>
<td>16</td>
<td>15</td>
<td>8</td>
<td>6</td>
<td>13</td>
<td>1</td>
<td>14</td>
<td>1</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>1</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>33</td>
<td>20</td>
<td>21</td>
<td>12</td>
<td>15</td>
<td>9</td>
<td>7</td>
<td>11</td>
<td>1</td>
<td>10</td>
<td>3</td>
<td>8</td>
<td>14</td>
<td>7</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>34</td>
<td>20</td>
<td>21</td>
<td>12</td>
<td>15</td>
<td>9</td>
<td>7</td>
<td>11</td>
<td>1</td>
<td>10</td>
<td>3</td>
<td>9</td>
<td>14</td>
<td>5</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>35</td>
<td>18</td>
<td>22</td>
<td>13</td>
<td>10</td>
<td>9</td>
<td>5</td>
<td>20</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>7</td>
<td>12</td>
<td>4</td>
<td>11</td>
<td>0</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>36</td>
<td>18</td>
<td>12</td>
<td>21</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>18</td>
<td>0</td>
<td>13</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>12</td>
<td>0</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>37</td>
<td>17</td>
<td>12</td>
<td>22</td>
<td>10</td>
<td>8</td>
<td>5</td>
<td>18</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>5</td>
<td>13</td>
<td>4</td>
<td>10</td>
<td>0</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>38</td>
<td>17</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>8</td>
<td>6</td>
<td>15</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>9</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>11</td>
</tr>
<tr>
<td>39</td>
<td>19</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>7</td>
<td>6</td>
<td>15</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>40</td>
<td>18</td>
<td>16</td>
<td>16</td>
<td>19</td>
<td>8</td>
<td>6</td>
<td>11</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>41</td>
<td>18</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>8</td>
<td>6</td>
<td>15</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>8</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>42</td>
<td>15</td>
<td>12</td>
<td>14</td>
<td>23</td>
<td>8</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td>9</td>
<td>4</td>
<td>7</td>
<td>15</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>11</td>
</tr>
<tr>
<td>43</td>
<td>13</td>
<td>11</td>
<td>14</td>
<td>22</td>
<td>7</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td>10</td>
<td>4</td>
<td>8</td>
<td>13</td>
<td>6</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>44</td>
<td>16</td>
<td>11</td>
<td>15</td>
<td>23</td>
<td>10</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td>10</td>
<td>3</td>
<td>7</td>
<td>12</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>45</td>
<td>14</td>
<td>11</td>
<td>15</td>
<td>24</td>
<td>11</td>
<td>3</td>
<td>19</td>
<td>2</td>
<td>10</td>
<td>2</td>
<td>7</td>
<td>12</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>46</td>
<td>11</td>
<td>11</td>
<td>15</td>
<td>24</td>
<td>10</td>
<td>5</td>
<td>18</td>
<td>1</td>
<td>11</td>
<td>1</td>
<td>7</td>
<td>14</td>
<td>5</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>47</td>
<td>15</td>
<td>9</td>
<td>12</td>
<td>21</td>
<td>8</td>
<td>4</td>
<td>21</td>
<td>1</td>
<td>10</td>
<td>3</td>
<td>7</td>
<td>15</td>
<td>7</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>48</td>
<td>15</td>
<td>11</td>
<td>15</td>
<td>22</td>
<td>7</td>
<td>3</td>
<td>19</td>
<td>1</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>14</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>10</td>
</tr>
</tbody>
</table>
PCA for $x_1, \ldots, x_n \in \mathbb{R}^m$:

$$\arg\min_{\psi, \phi_1, \ldots, \phi_n} \sum_{i=1}^{n} \|x_i - \psi \phi_i\|^2, \quad \phi_i \in \mathbb{R}^K, \psi \in \mathbb{R}^{m \times K}, K \leq n$$

$\Rightarrow$ SVD: for $X := (x_1, \ldots, x_n)^T \in \mathbb{R}^{n \times m}$:

$$X = UDV^T$$

with $U \in \mathbb{R}^{n \times m}$ orthonormal ($U^T U = I$)

$V \in \mathbb{R}^{m \times m}$ orthonormal ($V^T V = I$)

$D \in \mathbb{R}^{m \times m}$ diagonal with $D_{1,1} \geq D_{2,2} \geq \ldots \geq D_{m,m}$

yields

$$\psi = (V_{.,1}, \ldots, V_{.,K})$$

$$\phi_i = \psi^T x_i, \quad i = 1, \ldots, n$$

Solve SVD algebraically.
PCA

[Tipping and Bishop, 1999]
## Factorization Models for Recommender Systems and Other Applications

### 1. Matrix Factorization Models

#### PCA

<table>
<thead>
<tr>
<th></th>
<th>V1</th>
<th>V2</th>
<th>V3</th>
<th>V4</th>
<th>V5</th>
<th>V6</th>
<th>V7</th>
<th>V8</th>
<th>V9</th>
<th>V10</th>
<th>V11</th>
<th>V12</th>
<th>V13</th>
<th>V14</th>
<th>V15</th>
<th>V16</th>
<th>V17</th>
<th>V18</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>13</td>
<td>14</td>
<td>16</td>
<td>4</td>
<td>9</td>
<td>14</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>11</td>
<td>13</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>11</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>6</td>
<td>5</td>
<td>12</td>
<td>1</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>18</td>
<td>16</td>
<td>16</td>
<td>8</td>
<td>6</td>
<td>14</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td></td>
<td>11</td>
</tr>
<tr>
<td>14</td>
<td>18</td>
<td>16</td>
<td>15</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>7</td>
<td>13</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>13</td>
<td></td>
<td>22</td>
<td>8</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td></td>
<td>8</td>
<td>11</td>
<td>7</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>13</td>
<td>16</td>
<td>9</td>
<td>3</td>
<td>17</td>
<td>1</td>
<td></td>
<td>4</td>
<td>7</td>
<td>12</td>
<td></td>
<td>5</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>22</td>
<td>19</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>18</td>
<td>1</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td>11</td>
<td>6</td>
<td>8</td>
<td>1</td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
PCA with missing values

\[
\begin{align*}
\arg\min_{\psi, \phi_1, \ldots, \phi_n} \sum_{i=1}^{n} ||x_i - \Psi \phi_i||^2, & \quad \phi_i \in \mathbb{R}^K, \Psi \in \mathbb{R}^{m \times K}, K \leq n \\
= \arg\min_{\psi_1, \ldots, \psi_m, \phi_1, \ldots, \phi_n} \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{i,j} - \psi_j^T \phi_i)^2, & \quad \Psi = (\psi_1, \ldots, \psi_m) \\
\leadsto \arg\min_{\psi_1, \ldots, \psi_m, \phi_1, \ldots, \phi_n} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} (x_{i,j} - \psi_j^T \phi_i)^2, & \quad w_{i,j} \in \{0, 1\}
\end{align*}
\]

Solve numerically.
PCA

from complete data

from data with 20% missing values

[Tipping and Bishop, 1999]
The Rating Recommendation Problem

<table>
<thead>
<tr>
<th>item</th>
<th>Toy Story</th>
<th>GoldenEye</th>
<th>Four Rooms</th>
<th>Get Shorty</th>
<th>Copycat</th>
<th>Shanghai Triad</th>
<th>Twelve Monkeys</th>
<th>Babe</th>
<th>Dead Man Walking</th>
</tr>
</thead>
<tbody>
<tr>
<td>user Anna</td>
<td>4</td>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Berta</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clara</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daniel</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Egon</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Fred</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Recommendation Problem / Attributes

<table>
<thead>
<tr>
<th>gender</th>
<th>age</th>
<th>item</th>
<th>year</th>
<th>action</th>
<th>children's</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>'95</td>
<td>−</td>
<td>+ − − − − − − − − − − − − + − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>'95</td>
<td>+</td>
<td>− − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>'95</td>
<td>−</td>
<td>+ − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>'95</td>
<td>+</td>
<td>− − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td>'95</td>
<td>−</td>
<td>+ − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td>'95</td>
<td>+</td>
<td>− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td>'95</td>
<td>−</td>
<td>+ − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td>'95</td>
<td>+</td>
<td>− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9</td>
<td>'95</td>
<td>−</td>
<td>+ − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
<td>'95</td>
<td>+</td>
<td>− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −</td>
</tr>
</tbody>
</table>

User 1: m 24
User 2: f 53
User 3: m 23
User 4: m 24
User 5: f 33
User 6: m 42

Discard user/item attributes (for a start).
Is Rating Prediction a Regression Problem?

- Yes, one for each user, with the item attributes as predictors (content-based filtering),
  - but then we do **not share information between users** (collaborative information)
  - but we said we have **no item attributes** for a start.
Is Rating Prediction a Regression Problem?

- Yes, one for each user, with the item attributes as predictors (content-based filtering),
  - but then we do **not share information between users** (collaborative information)
  - but we said we have **no item attributes** for a start.

- Yes, one for each item, with the ratings for all the other items as predictors,
  - but there are lots of **missing values**,
  - but all these regression **problems are related** (multi-task learning).
Is Rating Prediction a Regression Problem?

- Yes, one for each user, with the item attributes as predictors (content-based filtering),
  - but then we do not share information between users (collaborative information)
  - but we said we have no item attributes for a start.

- Yes, one for each item, with the ratings for all the other items as predictors,
  - but there are lots of missing values,
  - but all these regression problems are related (multi-task learning).

- Yes, with just two predictors user ID and item ID,
  - but both variables are nominal with many levels, without interactions not leading to personalized models,
  - but with interactions having as many parameters as possible instances.
Collaborative Filtering (Nearest Neighbor Models)

Idea: let's use a similarity measure on the available data.

\[
\hat{y}(u, i) := \bar{y}_{u,.} + \frac{1}{|N_u|} \sum_{v \in N_u} (y_{v,i} - \bar{y}_{v,.}), \quad N_u := \text{argmax}_{v \in U} \sim(u, v)
\]

\[
\sim(u, v) := \frac{\langle y_{u|v}, y_{v|u} \rangle}{\|y_{u|v}\|_2 \|y_{v|u}\|_2}
\]

\[
y_{u|v} := (y_{u,i})_{i \in I_u \cap I_v}, \quad I_u := \{i \in I \mid y_{u,i} \neq .\}
\]

- but the choice of the similarity measure is arbitrary,
- but a nearest neighbor model is not a strong regression model.

[Goldberg et al., 1992; Resnick et al., 1994]
Matrix Factorization

**Idea:** learn latent features on the available data.

and assume a simple **bilinear reconstruction model**.
Matrix Factorization

Idea: learn latent features on the available data.

\[ \hat{y}(u, i) := a + b_u + c_i + \langle \phi_u, \psi_i \rangle, \quad a, b_u, c_i \in \mathbb{R}, \phi_u, \psi_i \in \mathbb{R}^K \]

Regularized Loglikelihood:

\[ f(\phi, \psi; y) := \sum_{(u,i,y) \in D} (y - \hat{y}(u, i))^2 + \lambda (\|\phi\|^2 + \|\psi\|^2) \]

- \( D := \{(u_1, i_1, y_1), \ldots, (u_N, i_N, y_N)\} \) sparse matrix data (in triples).
- can be easily learned by stochastic gradient descent.
- \( \lambda \in \mathbb{R}_0^+ \) and \( K \in \mathbb{N} \) are hyperparameters.
Matrix Factorization

Idea: learn latent features on the available data.

\[ \hat{y}(u, i) := a + b_u + c_i + \sum_{k=1}^{K} \phi_{u,k} \psi_{i,k}, \quad a, b_u, c_i \in \mathbb{R}, \phi_u, \psi_i \in \mathbb{R}^K \]

Regularized Loglikelihood:

\[ f(\phi, \psi; y) := \sum_{(u,i,y) \in D} (y - \hat{y}(u, i))^2 + \lambda(\|\phi\|^2 + \|\psi\|^2) \]

- \( D := \{(u_1, i_1, y_1), \ldots, (u_N, i_N, y_N)\} \) sparse matrix data (in triples).
- can be easily learned by stochastic gradient descent.
- \( \lambda \in \mathbb{R}_0^+ \) and \( K \in \mathbb{N} \) are hyperparameters.
Recommendation Quality (Matrix Factorization)

![Figure 4](image_url)

**Figure 4.** Matrix factorization models’ accuracy. The plots show the root-mean-square error of each of four individual factor models (lower is better). Accuracy improves when the factor model’s dimensionality (denoted by numbers on the charts) increases. In addition, the more refined factor models, whose descriptions involve more distinct sets of parameters, are more accurate. For comparison, the Netflix system achieves $\text{RMSE} = 0.9514$ on the same dataset, while the grand prize’s required accuracy is $\text{RMSE} = 0.8563$.

[Koren et al., 2009]
Matrix Factorization / Matrix Notation

\[ f(\Phi, \Psi; Y, W) := \| W \odot (Y - \Phi \Psi^T) \|^2 + \lambda (\| \Phi \|^2 + \| \Psi \|^2) \]

with

- \( Y \in \mathbb{R}^{U \times I} \) (partially) observed matrix
- \( W \in \{0, 1\}^{U \times I} \) indicator for observed cells
- \( \Phi \in \mathbb{R}^{U \times K} \) latent user/row features
- \( \Psi \in \mathbb{R}^{I \times K} \) latent item/column features
- \( \odot \) cell-wise / Hadamard product
Matrix Factorization / General Loss and Regularizer

\[ \hat{y}(u, i) := a + b_u + c_i + \langle \phi_u, \psi_i \rangle \]

\[ f(a, b, c, \phi, \psi; D) := \sum_{(u, i, y) \in D} \ell(y, \hat{y}(u, i)) + \Omega(b, c, \phi, \psi) \]

with

- \( U, I \in \mathbb{N} \) number of users/rows, items/columns
- \( K \in \mathbb{N} \) number of latent features
- \( D := \{(u_1, i_1, y_1), \ldots, (u_N, i_N, y_N)\} \) sparse matrix data (in triples)
Matrix Factorization / General Loss and Regularizer

\[ \hat{y}(u, i) := a + b_u + c_i + \langle \phi_u, \psi_i \rangle \]

\[ f(a, b, c, \phi, \psi; D) := \sum_{(u, i, y) \in D} \ell(y, \hat{y}(u, i)) + \Omega(b, c, \phi, \psi) \]

with (continued)

\[ a \in \mathbb{R} \quad \text{global offset} \]
\[ b \in \mathbb{R}^U \quad \text{user/row offset/bias} \]
\[ c \in \mathbb{R}^I \quad \text{item/column offset/bias} \]
\[ \phi \in \mathbb{R}^{U \times K} \quad \text{latent user/row features} \]
\[ \psi \in \mathbb{R}^{I \times K} \quad \text{latent item/column features} \]
\[ \ell: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \quad \text{loss} \]
\[ \Omega: \mathbb{R}^{U+I+UK+IK} \rightarrow \mathbb{R} \quad \text{regularization} \]
Matrix Factorization / General Loss and Regularizer

\[
\hat{y}(u, i) := a + \langle \phi_u, \psi_i \rangle
\]

\[
f(a, \phi, \psi; D) := \sum_{(u, i, y) \in D} \ell(y, \hat{y}(u, i)) + \Omega(\phi, \psi)
\]

with one latent feature dimension fixed

\[
\phi_u^1 := 1
\]

\[
\psi_i^2 := 1
\]

thus

\[
b_u := \phi_u^2
\]

\[
c_i := \psi_i^1
\]
Unified View on MF [Singh and Gordon, 2008a]

\[
\hat{y}(u, i) := g(\Phi^T_{u,i}, \Psi_{i,\cdot}) \\
\hat{Y} := (\hat{y}(u, i))_{u \in U, i \in I} \\
f(\Phi, \Psi; Y) := D(Y \parallel \hat{Y}, W) + \Omega(\Phi, \Psi), \quad \Phi, \Psi \in \mathcal{C}
\]

with

\[
Y \in \mathbb{R}^{U \times I} \quad \text{(partially) observed matrix} \\
W \in \{0, 1\}^{U \times I} \quad \text{indicator for observed cells} \\
\Phi \in \mathbb{R}^{U \times K} \quad \text{latent user/row features} \\
\Psi \in \mathbb{R}^{I \times K} \quad \text{latent item/column features} \\
g : \mathbb{R} \rightarrow \mathbb{R} \quad \text{prediction link} \\
\mathcal{C} \quad \text{constraints on latent features} \\
\mathcal{D} \quad \text{Bregman divergence} \\
\Omega \quad \text{regularization}
## Unified View on MF [Singh and Gordon, 2008a]

| Method         | \( \text{dom } X_{ij} \) | \( \text{Link } f(\theta) \) | Loss \( \mathcal{D}(X|\hat{X} = f(\Theta), W) \) | \( W_{ij} \) |
|----------------|---------------------------|---------------------------|-------------------------------|----------------|
| SVD            | \( \mathbb{R} \)         | \( \theta \)            | \( \|W \circ (X - \hat{X})\|_2^2 \) | 1              |
| W-SVD         | \( \mathbb{R} \)         | \( \theta \)            | \( \|W \circ (X - \hat{X})\|_2^2 \) | \( \geq 0 \)  |
| k-means       | \( \mathbb{R} \)         | \( \theta \)            | \( \sum_{ij} |W_{ij}(X_{ij} - \hat{X}_{ij})| \) | 1              |
| k-medians     | \( \mathbb{R} \)         | \( \theta \)            | \( \sum_{ij} |W_{ij}(X_{ij} - \hat{X}_{ij})| \) | \( \geq 0 \)  |
| \( \ell_1 \)-SVD | \( \mathbb{R} \)     | \( \theta \)            | \( \sum_{ij} W_{ij}(X_{ij} \log \frac{x_{ij}}{X_{ij}}) \) | 1              |
| pLSI          | \( 1 \circ X = 1 \)     | \( \theta \)            | \( \sum_{ij} W_{ij}(X_{ij} \log \frac{x_{ij}}{x_{ij}} + \Theta_{ij} - X_{ij}) \) | 1              |
| NMF           | \( \mathbb{R}_+ \)       | \( \theta \)            | \( \|W \circ (X - \hat{X})\|_2^2 \) | 1              |
| \( \ell_2 \)-NMF | \( \mathbb{R}_+ \)  | \( \theta \)            | \( \sum_{ij} W_{ij}(X_{ij} \log \frac{x_{ij}}{X_{ij}} + (1 - X_{ij}) \log \frac{1-x_{ij}}{1-X_{ij}}) \) | 1              |
| Logistic PCA  | \( \{0, 1\} \)         | \( (1 + e^{-\theta})^{-1} \) | \( \sum_{ij} W_{ij}(X_{ij} \log \frac{x_{ij}}{X_{ij}} + (1 - X_{ij}) \log \frac{1-x_{ij}}{1-X_{ij}}) \) | 1              |
| E-PCA         | many                     | many                     | decomposable Bregman \( D_F \) | 1              |
| G^2L^2M       | many                     | many                     | decomposable Bregman \( D_F \) | 1              |
| MMMF          | \( \{0, \ldots, R\} \) | min-loss                 | \( \sum_{r=1}^{R-1} \sum_{ij:x_{ij} \neq 0} W_{ij} \cdot h(\Theta_{ij} - B_{ir}) \) | 1              |
| Fast-MMMF     | \( \{0, \ldots, R\} \) | min-loss                 | \( \sum_{r=1}^{R-1} \sum_{ij:x_{ij} \neq 0} W_{ij} \cdot h_\gamma(\Theta_{ij} - B_{ir}) \) | 1              |
Unified View on MF [Singh and Gordon, 2008a]

<table>
<thead>
<tr>
<th>Method</th>
<th>Constraints $U$</th>
<th>Constraints $V$</th>
<th>Regularizer $\mathcal{R}(U, V)$</th>
<th>Algorithm(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVD</td>
<td>$U^TU = I$</td>
<td>$V^TV = \Lambda$</td>
<td>—</td>
<td>Gaussian Elimination,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Power Method</td>
</tr>
<tr>
<td>W-SVD</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>Gradient, EM</td>
</tr>
<tr>
<td>$k$-means</td>
<td></td>
<td>$V^TV = I$, $V_{ij} \in {0, 1}$</td>
<td>—</td>
<td>EM</td>
</tr>
<tr>
<td>$k$-medians</td>
<td></td>
<td>$V^TV = I$, $V_{ij} \in {0, 1}$</td>
<td>—</td>
<td>Alternating</td>
</tr>
<tr>
<td>$\ell_1$-SVD</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>Alternating</td>
</tr>
<tr>
<td>pLSI</td>
<td>$1^TU1 = 1$</td>
<td>$1^TV = 1$</td>
<td>—</td>
<td>EM</td>
</tr>
<tr>
<td>NMF</td>
<td>$U_{ij} \geq 0$</td>
<td>$V_{ij} \geq 0$</td>
<td>—</td>
<td>Multiplicative, EM</td>
</tr>
<tr>
<td>$\ell_2$-NMF</td>
<td>$U_{ij} \geq 0$</td>
<td>$V_{ij} \geq 0$</td>
<td>—</td>
<td>Multiplicative, Alternating</td>
</tr>
<tr>
<td>Logistic PCA</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>EM</td>
</tr>
<tr>
<td>E-PCA</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>Alternating</td>
</tr>
<tr>
<td>G$^2$L$^2$M</td>
<td>—</td>
<td>—</td>
<td>$|U|<em>{Fro}^2 + |V|</em>{Fro}^2$</td>
<td>Alternating (Subgradient, Newton)</td>
</tr>
<tr>
<td>MMMF</td>
<td>—</td>
<td>—</td>
<td>$tr(UV^T)$</td>
<td>Semidefinite Program</td>
</tr>
<tr>
<td>Fast-MMMF</td>
<td>—</td>
<td>—</td>
<td>$\frac{1}{2}(|U|<em>{Fro}^2 + |V|</em>{Fro}^2)$</td>
<td>Conjugate Gradient</td>
</tr>
</tbody>
</table>
A Short History of Factorization Models

1901 Pearson [1901] invents PCA.

1976 Wiberg [1976] generalizes PCA to PCA with missing values.

1999 Roweis [1998] and Tipping and Bishop [1999] provide a probabilistic interpretation for PCA.


2006 Funk [2006] and Bell et al. [2007] popularize matrix factorization as leading method in the Netflix prize.

2008 Singh and Gordon [2008a,b] systematize matrix factorization models
  - with different losses, feature constraints, etc. as well as
  - for several matrices.
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
Gradient Descent (GD)

\[ \theta^* := \arg\min_{\theta} f(\theta) \]

1. \( GD(f, \eta, \sigma^2, \epsilon) : \)
2. initialize \( \theta \sim \mathcal{N}(0, \sigma^2) \)
3. repeat
   4. \( \theta^{old} := \theta \)
   5. \( \theta := \theta - \eta \frac{\partial f}{\partial \theta}(\theta) \)
4. until \( \| \theta - \theta^{old} \| < \epsilon \)
7. return \( \theta \)
### Stochastic Gradient Descent (SGD)

\[
\theta^* := \arg\min_{\theta} f(\theta)
\]

\[
f(\theta) := \sum_{c \in C} f_c(\theta)
\]

1. \( \text{SGD}((f_c)_{c \in C}, \eta, \sigma^2, \epsilon) : \)
2. initialize \( \theta \sim \mathcal{N}(0, \sigma^2) \)
3. repeat
4. \( \theta^{old} := \theta \)
5. foreach \( c \in C \) in random order do
6. \( \theta := \theta - \eta \frac{\partial f_c}{\partial \theta}(\theta) \)
7. end
8. until \( \|\theta - \theta^{old}\| < \epsilon \)
9. return \( \theta \)
SGD to Learn a Model

\[ \theta^* := \arg\min_{\theta} f(\theta) \]

\[ f(\theta) := \sum_{(x,y) \in D} \ell(y, \hat{y}(x; \theta)) + \lambda \Omega(\theta) \]

1. `learn-SGD(\ell, \hat{y}, \Omega, \eta, \sigma^2, \epsilon)` :
2. initialize \( \theta \sim \mathcal{N}(0, \sigma^2) \)
3. repeat
4. \( \theta^{old} := \theta \)
5. foreach \( (x, y) \in D \) in random order do
6. \( \theta := \theta - \eta \left( \frac{\partial \ell}{\partial \hat{y}}(y, \hat{y}(x; \theta)) \frac{\partial \hat{y}}{\partial \theta}(x; \theta) + \frac{1}{|D|} \lambda \frac{\partial \Omega}{\partial \theta}(\theta) \right) \)
7. end
8. until \( ||\theta - \theta^{old}|| < \epsilon \)
9. return \( \theta \)
SGD to Learn a Matrix Factorization

E.g., for

\[ \ell(y, \hat{y}) := (y - \hat{y})^2 \]

\[ \Omega(\theta) := ||\theta||^2 \]

\[ \hat{y}(x = (u, i); \theta) := a + b_u + c_i + \langle \phi_u, \psi_i \rangle \]

one easily gets:

\[ \frac{\partial \ell}{\partial \hat{y}} (y, \hat{y}) = -2(y - \hat{y}) \]

\[ \frac{\partial \Omega}{\partial \theta} (\theta) = 2\theta \]
SGD to Learn a Matrix Factorization

E.g., for

\[ \ell(y, \hat{y}) := (y - \hat{y})^2 \]
\[ \Omega(\theta) := ||\theta||^2 \]

\[ \hat{y}(x = (u, i); \theta) := a + b_u + c_i + \sum_{k=1}^{K} \phi_{u,k} \psi_{i,k} \]

one easily gets:

\[ \frac{\partial \ell}{\partial \hat{y}}(y, \hat{y}) = -2(y - \hat{y}) \]
\[ \frac{\partial \Omega}{\partial \theta}(\theta) = 2\theta \]

\[ \frac{\partial \hat{y}}{\partial \theta}(x = (u, i); \theta) = \begin{cases} 
\psi_{i,k}, & \text{for } \theta = \phi_{u,k} \\
\phi_{u,k}, & \text{for } \theta = \psi_{i,k} \\
1, & \text{for } \theta \in \{a, b_u, c_i\} 
\end{cases} \]
SGD / Sparse Parameter Updates

When a component function $f_c$ depends only on a subset of the parameters $\theta$, only those parameters have to be updated.

For factorization models:

- $\ell(y, \hat{y}(x = (u, i); \theta))$ does only depend on $\phi_u, \psi_i$.
  (not on $\phi_{u'}, \psi_{i'}$ for $u' \neq u, i' \neq i$)

- Update only $\phi_u, \psi_i$. 
SGD / Sparse Parameter Updates

When a component function $f_c$ depends only on a subset of the parameters $\theta$, only those parameters have to be updated.

For factorization models:

- $\ell(y, \hat{y}(x = (u, i); \theta))$ does only depend on $\phi_u, \psi_i$.  
  (not on $\phi_{u'}, \psi_{i'}$ for $u' \neq u, i' \neq i$)
  - update only $\phi_u, \psi_i$.

- but beware: the regularization term depends on all parameters!
  - update whenever we have to touch the parameter anyway, e.g.,

$$
\phi_{u,k} := \phi_{u,k} - \eta \left( -2(y - \hat{y}(x; \theta))\psi_{i,k} + \frac{1}{|D|} \frac{|D|}{\text{freq}(u; D)} \lambda^2 \phi_{u,k} \right)
$$

$$
\text{freq}(u; D) := |\{(u', i', y') \in D \mid u' = u\}|
$$

- this normalization often is missing, leading to stronger regularization for parameters involved in more components.
Block Coordinate Descent

\[ \theta^* := \arg\min_{\theta} f(\theta) \]

\[ \theta \in \Theta := \prod_{c \in C} \Theta_c \]

Denote the function with parameters outside block \( c \) fixed to \( \theta \) by

\[ f|_{\Theta_c, \theta}(\tilde{\theta}_c) := f(\theta|_{\prod_{c' < c} \Theta_{c'}, \tilde{\theta}_{c'}, \theta|_{\prod_{c' > c} \Theta_{c'}}}, \tilde{\theta}_c) \in \Theta_c \]

1. \( BCD(f, (\Theta_c)_{c \in C}, \sigma^2, \epsilon) : \)
2. initialize \( \theta \sim \mathcal{N}(0, \sigma^2) \)
3. repeat
   4. \( \theta^{old} := \theta \)
   5. foreach \( c \in C \) in random order do
      6. \( \theta|_{\Theta_c} := \arg\min_{\tilde{\theta}_c} f|_{\Theta_c, \theta}(\tilde{\theta}_c) \)
   7. end
5. until \( ||\theta - \theta^{old}|| < \epsilon \)
6. return \( \theta \)
Example for Fixing Function Arguments

\[ f(\theta_1, \theta_2, \theta_3) := 10 + \theta_1 + \theta_2 \cdot \theta_3 \]
\[ \Theta := \mathbb{R}^3 \]
Example for Fixing Function Arguments

\[ f(\theta_1, \theta_2) := f(\theta_{1,1}, \theta_{1,2}, \theta_2) := 10 + \theta_{1,1} + \theta_{1,2} \cdot \theta_2 \]

\[ \Theta := \mathbb{R}^3 \]

\[ \Theta_1 := \mathbb{R}^2, \quad \Theta_2 := \mathbb{R} \]

\[ \theta := ((3, 7), -5) \]

\[ f|_{\Theta_1, \theta}(\tilde{\theta}_1) := f(\theta|_{\prod c'_{<1} \Theta_{c'}}, \tilde{\theta}_1, \theta|_{\prod c'_{>1} \Theta_{c'}}), \quad \tilde{\theta}_1 \in \Theta_1 = \mathbb{R}^2 \]

\[ = f(\tilde{\theta}_1, \theta_2) \]

\[ = f(\tilde{\theta}_{1,1}, \tilde{\theta}_{1,2}, -5) = 10 + \tilde{\theta}_{1,1} + \tilde{\theta}_{1,2} \cdot 5 \]
Example for Fixing Function Arguments

\[ f(\theta_1, \theta_2) := f(\theta_{1,1}, \theta_{1,2}, \theta_2) := 10 + \theta_{1,1} + \theta_{1,2} \cdot \theta_2 \]

\[ \Theta := \mathbb{R}^3 \]

\[ \Theta_1 := \mathbb{R}^2, \quad \Theta_2 := \mathbb{R} \]

\[ \theta := ((3, 7), -5) \]

\[ f|_{\Theta_1, \theta}(\tilde{\theta}_1) := f(\theta|_{\prod_{c' < 1} \Theta_{c'}}, \tilde{\theta}_1, \theta|_{\prod_{c' > 1} \Theta_{c'}}), \quad \tilde{\theta}_1 \in \Theta_1 = \mathbb{R}^2 \]

\[ = f(\tilde{\theta}_1, \theta_2) \]

\[ = f(\tilde{\theta}_{1,1}, \tilde{\theta}_{1,2}, -5) = 10 + \tilde{\theta}_{1,1} + \tilde{\theta}_{1,2} \cdot 5 \]

\[ f|_{\Theta_2, \theta}(\tilde{\theta}_2) := f(\theta|_{\prod_{c' < 1} \Theta_{c'}}, \tilde{\theta}_2, \theta|_{\prod_{c' > 1} \Theta_{c'}}), \quad \tilde{\theta}_2 \in \Theta_2 = \mathbb{R} \]

\[ = f(\theta_1, \tilde{\theta}_2) \]

\[ = f(3, 7, \tilde{\theta}_2) = 10 + 3 + 7 \cdot \tilde{\theta}_2 = 13 + 7 \cdot \tilde{\theta}_2 \]
Block Coordinate Descent

Useful if the component problems

$$\arg\min_{\tilde{\theta}_c} f|_{\Theta_c, \theta(\tilde{\theta}_c)}$$

are simpler to solve.

For matrix factorization:

$$\Theta_U := (\{(b_u)_{u \in U}, (\phi_u,)_{u \in U}\})$$

$$\Theta_I := (\{(c_i)_{i \in I}, (\psi_i,)_{i \in I}\})$$

Then the component problems are just regression problems:

$$\theta_U | \theta_I := \arg\min_{b_u, \phi_u} \sum_{(u, i, y) \in D} \ell(y, a + b_u + c_i + \langle \phi_u, \psi_i \rangle) + \lambda \Omega(\theta_U)$$

$$\theta_I | \theta_U := \arg\min_{c_i, \psi_i} \sum_{(u, i, y) \in D} \ell(y, a + b_u + c_i + \langle \phi_u, \psi_i \rangle) + \lambda \Omega(\theta_I)$$
Block Coordinate Descent

Especially for $L_2$ loss and $L_2$ regularization: ridge regression.

$$\theta^* := \arg\min_{\theta} \sum_{(x,y) \in D} (y - \langle x, \theta \rangle)^2 + \lambda \|\theta\|^2$$

Could be solved by solving a linear system of equations:

$$(X^T X + \lambda I) \theta^* = X^T Y,$$

with $X := (x_1^T, \ldots, x_N^T)^T$, $Y := (y_1, \ldots, y_N)^T$, $D := \{(x_1, y_1), \ldots, (x_N, y_N)\}$

This specific block coordinate descent is called **Alternating Least Squares** in the literature [Bell and Koren, 2007].
Coordinate Descent

Often for coordinate descent blocks of dimension 1 (scalars) are best, esp. if the 1-dimensional component problem can be solved analytically.

For 1-dimensional ridge-regression:

\[ \theta^* = (X^T X + \lambda I)^{-1} (X^T Y) = \frac{\sum_{i=1}^{N} x_i y_i}{\sum_{i=1}^{N} x_i^2 + \lambda} \]

Coordinate descent for matrix factorization models with L2 loss and L2 regularization is called ALS1 [Pilászy et al., 2010].
Coordinate Descent (ALS1) / Convergence Speed

[Plot showing the convergence speed of ALS and ALS1]

[Pilászy et al., 2010]
Damped Newton Algorithm

1. \( \text{Newton}(f, \eta, \sigma^2, \epsilon) : \)
2. initialize \( \theta \sim \mathcal{N}(0, \sigma^2) \)
3. repeat
   4. \( \theta^{old} := \theta \)
   5. \( d := \text{solve}_d \left( \frac{\partial^2 f}{\partial \theta^2}(\theta) + \eta I \right) \)
   6. \( d = -\frac{\partial f}{\partial \theta}(\theta) \)
   7. \( \theta := \theta + d \)
   8. until \( \| \theta - \theta^{old} \| < \epsilon \)
9. return \( \theta \)
Newton for Factorization Models

\[ f(\Phi, \Psi; Y) := \|W \odot (Y - \Phi \Psi^T)\|^2 + \lambda(\|\Phi\|^2 + \|\Psi\|^2) \]

\[
\frac{\partial^2 f}{\partial \phi_u \partial \phi_v} = 2\delta(u = v)(\Psi^T \text{diag}(W_{u,\cdot})^2 \Psi + 2\lambda I)
\]

\[
\frac{\partial^2 f}{\partial \phi_u \partial \psi_i} = 2w_{u,i}(\phi_u \psi_i^T + I(\phi_u^T \psi_i - Y_{u,i}))
\]

\[
\frac{\partial^2 f}{\partial \psi_i \partial \psi_j} = 2\delta(i = j)(\Phi^T \text{diag}(W_{\cdot,i})^2 \Phi + 2\lambda I)
\]

[Buchanan and Fitzgibbon, 2005]

Notation:
\[ \delta(A) := \begin{cases} 
1, & \text{if } A \text{ is true} \\
0, & \text{else} 
\end{cases} \]
Newton for Factorization Models
e.g., for a small $72 \times 319$ matrix with 28% given values:

[Buchanan and Fitzgibbon, 2005]
Newton for Factorization Models / Variants

FNMA Fast Nonnegative Matrix Approximation [Kim et al., 2007]:

- alternating least squares
- solving each component problem via a Quasi-Newton method (BFGS).

Stochastic Newton [Singh and Gordon, 2008b]:

- alternating least squares
- solving each component problem via a stochastic Quasi-Newton method.

LM+ [Phan et al., 2011]:

- extension to tensors.
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
Two Different Recommendation Tasks

**predict rating values / votes:**

<table>
<thead>
<tr>
<th>item</th>
<th>Toy Story</th>
<th>GoldenEye</th>
<th>Four Rooms</th>
<th>Get Shorty</th>
<th>Shanghai Triad</th>
<th>Twelve Monkeys</th>
<th>Babe</th>
<th>Dead Man</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>user Anna</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Berta</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clara</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daniel</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Egon</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fred</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

▶ model votes of non-rated items as missing values?

**predict (rated) items:**

<table>
<thead>
<tr>
<th>item</th>
<th>Toy Story</th>
<th>GoldenEye</th>
<th>Four Rooms</th>
<th>Get Shorty</th>
<th>Shanghai Triad</th>
<th>Twelve Monkeys</th>
<th>Babe</th>
<th>Dead Man</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>user Anna</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Berta</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clara</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daniel</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Egon</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fred</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

▶ there are only positive, no (explicit) negative examples.
Weighted Regularized Matrix Factorization

idea: assign a small weight to the negative examples.

\[
\hat{y}(u, i) := \langle \phi_u, \psi_i \rangle, \quad \phi_u, \psi_i \in \mathbb{R}^K
\]

\[
f(\phi, \psi) := \sum_{(u, i) \in U \times I} w_{u,i}(\delta((u, i) \in D) - \hat{y}(u, i))^2 + \lambda(\|\phi\| + \|\psi\|)
\]

<table>
<thead>
<tr>
<th>(w_{u,i}) (for ((u, i) \in D))</th>
<th>(w_{u,i}) (for ((u, i) \notin D))</th>
<th>reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 + (\alpha r_{u,i})</td>
<td>1</td>
<td>[Hu et al., 2008]</td>
</tr>
<tr>
<td>1</td>
<td>(\alpha)</td>
<td>[Pan et al., 2008]</td>
</tr>
<tr>
<td>1</td>
<td>(\alpha n_u)</td>
<td>[Pan et al., 2008]</td>
</tr>
<tr>
<td>1</td>
<td>(\alpha(U - n_i))</td>
<td>[Pan et al., 2008]</td>
</tr>
</tbody>
</table>

where

- \(D \subseteq U \times I\) is the training data,
- \(r_{u,i}\) (additionally observed) ratings,
- \(n_u := |\{(u', i') \in D \mid u = u'\}|\) number of events for a user,
- \(\alpha\) a hyperparameter (e.g., \(\alpha = 40\) in [Hu et al., 2008]).
WR-MF / Recommendation Quality

(a) Yahoo news data

(b) User-Tag data

[Pan et al., 2008]
### BPR / Data Interpretation

**Absolute preference:**

<table>
<thead>
<tr>
<th>Item</th>
<th>User</th>
<th>Item</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>u_1</td>
<td>?</td>
<td>i_1</td>
<td>?</td>
</tr>
<tr>
<td>u_2</td>
<td>+</td>
<td>i_2</td>
<td>+</td>
</tr>
<tr>
<td>u_3</td>
<td>+</td>
<td>i_3</td>
<td>+</td>
</tr>
<tr>
<td>u_4</td>
<td>?</td>
<td>i_4</td>
<td>+</td>
</tr>
<tr>
<td>u_5</td>
<td>?</td>
<td>i_5</td>
<td>?</td>
</tr>
</tbody>
</table>

\[ S \subseteq U \times I \]
BPR / Data Interpretation

**Absolute preference:**

\[
S \subseteq U \times I
\]

\[
D_S := \{(u, i, j) \in U \times I \times I \mid (u, i) \in S, (u, j) \notin S\}
\]
Bayesian Personalized Ranking (BPR)

Relative preferences, modeled by matrix factorization:

\[
\hat{p}(i > u j) := \sigma(\hat{y}_{u, i, j}) = \sigma(\hat{y}_{u, i} - \hat{y}_{u, j}) = \sigma(\langle \phi_u, \psi_i \rangle - \langle \phi_u, \psi_j \rangle)
\]

\[
\sigma(x) := \frac{1}{1 + e^{-x}}
\]

Regularized Loglikelihood:

\[
\text{BPR-Opt}(\Theta; D_s) := - \sum_{(u, i, j) \in D_s} \ln \hat{p}(i > u j) + \lambda_\phi \| \phi \|^2 + \lambda_\psi \| \psi \|^2
\]

with parameters \( \Theta := (\phi, \psi) \).
BPR / Learning BPR via SGD

1. LearnBPR($D_S, \eta, \lambda$):
2. initialize $\Theta := (\phi, \psi)$
3. repeat
4. draw $(u, i, j)$ from $D_S$
5. $\Theta \leftarrow \Theta + \eta \left( \frac{e^{-\hat{y}_{u,i,j}}}{1+e^{-\hat{y}_{u,i,j}}} \cdot \frac{\partial}{\partial \Theta} \hat{y}_{u,i,j} - \lambda \Theta \cdot \Theta \right)$
6. until convergence
7. Return $\Theta$

\[
\frac{\partial}{\partial \phi_u} \hat{y}_{u,i,j} = \psi_i - \psi_j
\]
\[
\frac{\partial}{\partial \psi_i} \hat{y}_{u,i,j} = \phi_u
\]
\[
\frac{\partial}{\partial \psi_j} \hat{y}_{u,i,j} = -\phi_u
\]
BPR / Evaluation

Datasets

| dataset      | users $|U|$ | items $|I|$ | transactions $|S|$ |
|--------------|-------|-------|------------|
| Rossmann     | 10,000| 4000  | 426,612     |
| Netflix sample | 10,000| 5000  | 565,738     |

Evaluated methods

- SVD-MF
- WR-MF [Pan et al., 2008; Hu et al., 2008]
- nearest neighbor with cosine similarity
- constant (most popular)
- upper bound on AUC for non-personalized models

Netflix sample: target: rating event; every user and item occurs in at least 10 transactions.
BPR / Recommendation Quality

Video Rental: Netflix

Online shopping: Rossmann

Also see Ning and Karypis [2011] for an evtl. improved model (Sparse Linear Method, SLIM).

[Rendle et al., 2009]
Ordinal MF

Targets often are ordinal.

- i.e., ratings not continuous in $[1, 5]$, but discrete in $R := \{1, 2, 3, 4, 5\}$.

goals:

- improved rating prediction
  - use standard regression evaluation measures:

  $$
  \text{RMSE}(D, \hat{r}) := \left( \frac{1}{|D|} \sum_{(u,i,y) \in D} (y - \hat{r}(u, i))^2 \right)^{\frac{1}{2}}
  $$

  $$
  \text{MAE}(D, \hat{r}) := \frac{1}{|D|} \sum_{(u,i,y) \in D} |y - \hat{r}(u, i)|
  $$

- better fit to application
  - optimize ranking evaluation measures:

  $$
  \text{FCP}(D, \hat{y}) := \frac{\sum_{(u,i,y),(u,i',y') \in D} \delta(y > y') \delta(\hat{y}(u, i) > \hat{y}(u, i'))}{\sum_{(u,i,y),(u,i',y') \in D} \delta(y > y')} \cdot \frac{1}{|D|} \sum_{(u,i,y) \in D} |y - \hat{r}(u, i)|
  $$
Ordinal MF / Maximum Margin MF (MMMF)

idea: push score $\hat{y}$ into ordinal level thresholds $\rho$ with margin 1:

$$\infty =: \rho_0 < \rho_1 < \ldots < \rho_{R-1} < \rho_R := \infty$$

$$\rho_{y-1} + 1 \leq \hat{y} \leq \rho_y - 1$$

$$\rho_r + 1 \leq \hat{y}, \quad \text{for } r < y \quad \leadsto [1 - (\hat{y} - \rho_r)]_+ \min.$$  

$$\hat{y} \leq \rho_y - 1 \quad \text{for } r \geq y \quad \leadsto [1 - (\rho_r - \hat{y})]_+ \min.$$  

summation yields the ordinal MMMF loss:

$$\ell(y, \hat{y}; \rho) := \sum_{r=1}^{R-1} [1 - (-1)^{y>r}(\rho_r - \hat{y})]_+$$

$$\hat{r}(\hat{y}; \rho) := \min \{ r \in R \mid \hat{y} < \rho_r \}$$

[Rennie and Srebro, 2005]
Ordinal MF / OrdRec

similar idea as MMMF, but just using the most strict thresholds:

\[ \rho_{y-1} \leq \hat{y} \quad \Rightarrow \rho_{y-1} - \hat{y} \min. \]
\[ \hat{y} \leq \rho_y \quad \Rightarrow \hat{y} - \rho_y \min. \]

Wrapping terms with a sigmoid and summing yields the OrdRec loss:

\[ \ell(y, \hat{y}; \rho) := \sigma(\rho_{y-1} - \hat{y}) + \sigma(\hat{y} - \rho_y) \]

To ensure strict monotonicity of \( \rho \), the parametrization \((\rho_0, \alpha_1, \ldots, \alpha_{R-2})\) is used with \( \rho_{r+1} := \rho_r + e^{\alpha_r} \).

[Koren and Sill, 2011]

The same ordinal model is used by Paquet et al. [2012] (but with the probit instead of the logistic function and embedded in a hierarchical Bayesian model).
Ranking

- some ranking measures decompose into pairwise comparisons
  - e.g., Fraction of Concordant Pairs (FCP):
    \[
    \text{FCP}(D, \hat{y}) := \frac{\sum_{(u,i,y),(u,i',y') \in D} \delta(y > y')\delta(\hat{y}(u,i) > \hat{y}(u,i'))}{\sum_{(u,i,y),(u,i',y') \in D} \delta(y > y')}
    \]
- others are not decomposable
  - e.g., Normalized Discounted Cumulative Gain at \(k\) (NDCG\@\(k\)):
    \[
    \text{DCG}_k(y, \pi) := \sum_{j=1}^{k} \frac{2y_{\pi(j)} - 1}{\log_2(j + 1)}
    \]
    \[
    \text{NDCG}_k(y, \pi) := \frac{\text{DCG}_k(y, \pi)}{\text{DCG}_k(y, \text{argsort} y)}
    \]
    \[
    \text{NDCG}_k(D, \hat{y}) := \frac{1}{U} \sum_{u=1}^{U} \text{NDCG}_k(D_u, \text{argsort} \hat{y}(u, .))
    \]
Ranking / CoFiRank-ordinal

idea: use a pairwise loss with margin 1.

\[ \hat{y}_j + 1 \leq \hat{y}_i, \quad \text{for } y_j < y_i \leadsto [1 - (\hat{y}_j - \hat{y}_i)]_+ \min. \]

averaging yields the CoFiRank ordinal loss:

\[ \ell(y, \hat{y}) := \frac{2}{|I_u|^2 - \sum_{r \in R} |I_r|^2} \sum_{i,j \in I_u: y_i > y_j} c(y_i, y_j)[1 - (\hat{y}_i - \hat{y}_j)]_+ \]

where

- \( y, \hat{y} \in R^I \) are vectors of ratings for all items by a user and
- \( c(y, y') \) is the cost for confusing the ranks of items at levels \( y, y' \).
- CofiRank is a ranking model and does not provide rating predictions \( \hat{r} \).

Notation:

- \( I_u := \{ i \in I \mid (u, i, y) \in D \} \) are all items actually rated by a user,
- \( I_u^r := \{ i \in I \mid (u, i, y) \in D, y = r \} \) are all items actually rated as \( r \) by a user.

[Weimer et al., 2008a,b]
Ranking / CoFiRank-NDCG

\[ \ell(y, \hat{y}; c) := \max_{\pi} \left( 1 - \text{NDCG}(y, \pi) + \langle c, \hat{y}^{\pi} - \hat{y} \rangle \right) \]

- convex upper bound for
  \[ 1 - \text{NDCG}(y, \text{argsort } \hat{y}) \]
- but non-decomposable
  - requires the structured prediction framework and a special learning algorithm (see Weimer et al. [2008a] for details).

Notation:
\[ y^{\pi} := (y^{\pi}(1), y^{\pi}(2), \ldots, y^{\pi}(n)), \quad y \in \mathbb{R}^n, \pi \in S_n \]
Weimer et al. [2008a] use \[ c_i := (i + 1)^{-0.25}. \]
### Ranking / CoFiRank

<table>
<thead>
<tr>
<th>Method</th>
<th>N=10</th>
<th>N=20</th>
<th>N=50</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EachMovie</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoFi\textsuperscript{RANK} - NDCG</td>
<td>0.6367 ± 0.001</td>
<td>0.6619 ± 0.0022</td>
<td>0.6771 ± 0.0019</td>
</tr>
<tr>
<td>GPR</td>
<td>0.4558 ± 0.015</td>
<td>0.4849 ± 0.0066</td>
<td>0.5375 ± 0.0089</td>
</tr>
<tr>
<td>CGPR</td>
<td>0.5734 ± 0.014</td>
<td>0.5989 ± 0.0118</td>
<td>0.6341 ± 0.0114</td>
</tr>
<tr>
<td>GPOR</td>
<td>0.3692 ± 0.002</td>
<td>0.3678 ± 0.0030</td>
<td>0.3663 ± 0.0024</td>
</tr>
<tr>
<td>CGPOR</td>
<td>0.3789 ± 0.011</td>
<td>0.3781 ± 0.0056</td>
<td>0.3774 ± 0.0041</td>
</tr>
<tr>
<td>MMMF</td>
<td>0.4746 ± 0.034</td>
<td>0.4786 ± 0.0139</td>
<td>0.5478 ± 0.0211</td>
</tr>
<tr>
<td><strong>MovieLens</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoFi\textsuperscript{RANK} - NDCG</td>
<td>0.6237 ± 0.0241</td>
<td>0.6711 ± 0.0065</td>
<td>0.6455 ± 0.0103</td>
</tr>
<tr>
<td>GPR</td>
<td>0.4937 ± 0.0108</td>
<td>0.5020 ± 0.0089</td>
<td>0.5088 ± 0.0141</td>
</tr>
<tr>
<td>CGPR</td>
<td>0.5101 ± 0.0081</td>
<td>0.5249 ± 0.0073</td>
<td>0.5438 ± 0.0063</td>
</tr>
<tr>
<td>GPOR</td>
<td>0.4988 ± 0.0035</td>
<td>0.5004 ± 0.0046</td>
<td>0.5011 ± 0.0051</td>
</tr>
<tr>
<td>CGPOR</td>
<td>0.5053 ± 0.0047</td>
<td>0.5089 ± 0.0044</td>
<td>0.5049 ± 0.0035</td>
</tr>
<tr>
<td>MMMF</td>
<td>0.5521 ± 0.0183</td>
<td>0.6133 ± 0.0180</td>
<td><strong>0.6651 ± 0.0190</strong></td>
</tr>
</tbody>
</table>

Evaluation measure: NDCG@10. \(N\) is number of ratings/test user. GPR, CGPR, GPOR, CGPOR are Gaussian Process Models from Yu et al. [2006].

[Weimer et al., 2008a]
## Ranking / CoFiRank

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>N=10</th>
<th>N=20</th>
<th>N=50</th>
</tr>
</thead>
<tbody>
<tr>
<td>EachMovie</td>
<td>CoFi\textsuperscript{RANK} - NDCG</td>
<td>0.6562 ± 0.0012</td>
<td>0.6644 ± 0.0024</td>
<td>0.6406 ± 0.0040</td>
</tr>
<tr>
<td></td>
<td>CoFi\textsuperscript{RANK} - Ordinal</td>
<td><strong>0.6727 ± 0.0309</strong></td>
<td><strong>0.7240 ± 0.0018</strong></td>
<td><strong>0.7214 ± 0.0076</strong></td>
</tr>
<tr>
<td></td>
<td>CoFi\textsuperscript{RANK} - Regression</td>
<td>0.6114 ± 0.0217</td>
<td>0.6400 ± 0.0354</td>
<td>0.5693 ± 0.0428</td>
</tr>
<tr>
<td>MovieLens</td>
<td>CoFi\textsuperscript{RANK} - NDCG</td>
<td>0.6400 ± 0.0061</td>
<td>0.6307 ± 0.0062</td>
<td>0.6076 ± 0.0077</td>
</tr>
<tr>
<td></td>
<td>CoFi\textsuperscript{RANK} - Ordinal</td>
<td>0.6233 ± 0.0039</td>
<td>0.6686 ± 0.0058</td>
<td><strong>0.7169 ± 0.0059</strong></td>
</tr>
<tr>
<td></td>
<td>CoFi\textsuperscript{RANK} - Regression</td>
<td><strong>0.6420 ± 0.0252</strong></td>
<td>0.6509 ± 0.0190</td>
<td>0.6584 ± 0.0187</td>
</tr>
<tr>
<td></td>
<td>MMMF\textsuperscript{E}</td>
<td>0.6061 ± 0.0037</td>
<td><strong>0.6937 ± 0.0039</strong></td>
<td>0.6989 ± 0.0051</td>
</tr>
<tr>
<td>Netflix</td>
<td>CoFi\textsuperscript{RANK} - NDCG</td>
<td>0.6081</td>
<td>0.6204</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CoFi\textsuperscript{RANK} - Regression</td>
<td>0.6082</td>
<td>0.6287</td>
<td></td>
</tr>
</tbody>
</table>

Evaluation measure: NDCG@10. $N$ is number of ratings/test user.

[Weimer et al., 2008a]
Unary and Ordinal Targets / Summary

Two different problems:

- ordinal regression \(\rightsquigarrow\) models with ordinal binning losses.
  - bin thresholds are parameters that can be learned along with the other parameters.
  - MMMF, OrdRec, and the model in Paquet et al. [2012].

- ranking \(\rightsquigarrow\) models with pair losses or non-decomposable losses.
  - no additional parameters.
  - models with pair contrast can be learned by SGD etc., models with more complex losses require more complex learning algorithms.
  - unary targets usually handled as ranking problem.
  - CoFiRank, BPR.
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
Example: RS with Attributes

<table>
<thead>
<tr>
<th>gender</th>
<th>age</th>
<th>item 1</th>
<th>item 2</th>
<th>item 3</th>
<th>item 4</th>
<th>item 5</th>
<th>item 6</th>
<th>item 7</th>
<th>item 8</th>
<th>item 9</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>user 1</td>
<td>m</td>
<td>24</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>f</td>
<td>53</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>m</td>
<td>23</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>m</td>
<td>24</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>f</td>
<td>33</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>m</td>
<td>42</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Can we build better models using the attributes?
Example: RS with Item Attributes

Idea: factorize both matrices

\[
\hat{y}(u, i) := \langle \phi_u^Y, \psi_i^Y \rangle
\]
\[
\hat{z}(i, a) := \langle \phi_i^Z, \psi_a^Z \rangle
\]

\[
f(\phi^Y, \psi^Y, \phi^Z, \psi^Z) := \sum_{(u, i, y) \in D^Y} \ell^Y(y, \hat{y}(u, i)) + \sum_{(i, a, z) \in D^Z} \ell^Z(z, \hat{z}(i, a))
\]
\[
+ \Omega(\phi^Y, \psi^Y, \phi^Z, \psi^Z)
\]
Example: RS with Item Attributes

Idea: factorize both matrices \textit{sharing features}

\[
\hat{y}(u, i) := \langle \phi_u^Y, \psi_i^Y \rangle \\
\hat{z}(i, a) := \langle \phi_i^Z, \psi_a^Z \rangle \\
\psi_i^Y = \phi_i^Z
\]

\[
f(\phi^Y, \psi^Y, \phi^Z, \psi^Z) := \sum_{(u,i,y)\in D^Y} \ell^Y(y, \hat{y}(u, i)) + \sum_{(i,a,z)\in D^Z} \ell^Z(z, \hat{z}(i, a)) + \Omega(\phi^Y, \psi^Y, \phi^Z, \psi^Z)
\]
Example: RS with Item Attributes

Idea: factorize both matrices sharing features

\[ \hat{y}(u, i) := \langle \phi_u^U, \phi_i^I \rangle \]
\[ \hat{z}(i, a) := \langle \phi_i^I, \phi_a^A \rangle \]
\[ f(\phi^U, \phi^I, \phi^A) := \sum_{(u,i,y) \in D^Y} \ell^Y(y, \hat{y}(u, i)) + \sum_{(i,a,z) \in D^Z} \ell^Z(z, \hat{z}(i, a)) + \Omega(\phi^U, \phi^I, \phi^A) \]
Example: RS with Item Attributes

Idea: factorize both matrices sharing features, weighting the auxiliary loss

\[ \hat{y}(u, i) := \langle \phi_u^U, \phi_i^I \rangle \]
\[ \hat{z}(i, a) := \langle \phi^I_i, \phi^A_a \rangle \]
\[ f(\phi^U, \phi^I, \phi^A) := \sum_{(u, i, y) \in D^Y} \ell^Y(y, \hat{y}(u, i)) + \alpha \sum_{(i, a, z) \in D^Z} \ell^Z(z, \hat{z}(i, a)) \]
\[ + \Omega(\phi^U, \phi^I, \phi^A) \]

Relation Z has a regularization effect on \( \phi^I \): such \( \phi^I \) are preferred that are useful to also reconstruct \( Z \).
Multi-relational Prediction Problem (Matrices)

Let

- \( \mathcal{E} \) be a set of entity classes, each \( E \in \mathcal{E} \) a set of entities,
- \( \mathcal{Z} \) be a set of matrices, each \( Z \in \mathcal{Z} \) a matrix with
  - dimensions \( s(Z) \in \mathcal{E}^2 \) and
  - observed cells \( Z \subseteq \text{dom}(Z) \times \mathbb{R} \), with \( \text{dom}(Z) := s(Z)_1 \times s(Z)_2 \).
- \( Y \in \mathcal{Z} \) called target matrix, \( Z \in \mathcal{Z}, Z \neq Y \) auxiliary matrix.

Sought is a model

\[
\hat{y} : \text{dom}(Y) \to \mathbb{R}
\]

with small prediction error for unseen test instances \( Y_{\text{test}} \subseteq \text{dom}(Y) \times \mathbb{R} \) of the target matrix:

\[
\ell(\hat{y}; Y_{\text{test}}) := \sum_{(x_1,x_2,y) \in Y_{\text{test}}} \ell^{Y}(y, \hat{y}(x_1, x_2))
\]
Multi-relational Prediction Problem (Tensors)

Let

- $\mathcal{E}$ be a set of entity classes, each $E \in \mathcal{E}$ a set of entities,
- $\mathcal{Z}$ be a set of tensors, each $Z \in \mathcal{Z}$ a tensor with
  - dimensions $s(Z) \in \mathcal{E}^*$ and
  - observed cells $Z \subseteq \text{dom}(Z) \times \mathbb{R}$, with $\text{dom}(Z) := \prod_{i=1}^{\left| s(Z) \right|} s(Z)_i$.
- $Y \in \mathcal{Z}$ called target tensor, $Z \in \mathcal{Z}$, $Z \neq Y$ auxiliary tensor.

Sought is a model

$$\hat{y} : \text{dom}(Y) \to \mathbb{R}$$

with small prediction error for unseen test instances $Y_{\text{test}} \subseteq \text{dom}(Y) \times \mathbb{R}$ of the target tensor:

$$\ell(\hat{y}; Y_{\text{test}}) := \sum_{(x_1, x_2, y) \in Y_{\text{test}}} \ell^Y(y, \hat{y}(x_1, x_2))$$
Multi-relational Prediction Problem

Matrices (and tensors) can be used to encode positive and negative examples of relations:

\[ \text{KB} \models R(x_1, x_2) \iff Z(x_1, x_2) = 1 \]
\[ \text{KB} \models \neg R(x_1, x_2) \iff Z(x_1, x_2) = 0 \]

In the same way certainty factors could be encoded.

When only positive relation facts are contained in the knowledge base (in recsys sometimes called implicit feedback),

- vanilla factorization models cannot work,
- but see section 3 on unary targets about what to do.
Multi-relational Factorization Model (Matrices)

\[ \hat{Z}(x) := \langle \phi_{x_1}^{s(Z)_1}, \phi_{x_2}^{s(Z)_2} \rangle, \quad \forall Z \in \mathcal{Z}, x \in \text{dom}(Z) \]

\[
f((\phi^E)_{E \in \mathcal{E}}; Z) := \sum_{Z \in \mathcal{Z}} \alpha_Z \sum_{(x,z) \in Z} \ell^Z(z, \hat{Z}(x)) + \Omega((\phi^E)_{E \in \mathcal{E}})
\]

- Usually \( \alpha_Y := 1 \), the other \( \alpha \)'s are hyperparameters.
- \( \ell^Y \) is given by the problem, \( \ell^Z, Z \neq Y \) can be chosen ("structural hyperparameter").

Called

- Collective Matrix Factorization [Singh and Gordon, 2008b] (also allows prediction link function and latent feature constraints).
- Multi-relational Matrix Factorization [Lippert et al., 2008] (without the \( \alpha \) weights).
Multi-relational Factorization Model / Learning

All learning algorithms for matrix factorization models can be carried over to multi-relational matrix factorization models:

- sequentially:
  - first learn auxiliary relations, then target relation.
  - beware: factorization of the auxiliary relations not informed by the target relation!
    - corresponds to regression on unsupervised dimensionality reduction (e.g., PCA regression).

- jointly:
  - alternate steps for different relations.
  - ensure all relations are converged.
Multi-relational Factorization Model / Learning

1. SGD-MRFM($\mathcal{Z}, \alpha, \eta, \sigma^2, n, \epsilon$):
2. initialize $\phi^E \sim \mathcal{N}(0, \sigma^2)$ for all $E \in \mathcal{E}$
3. repeat
   4. $\phi^{old} := \phi$
   5. for $i = 1, \ldots, n$ do
   6. for $Z \in \mathcal{Z}$ do
      7. $E := s(Z)_1, F := s(Z)_2$
      8. draw $(x, y) \in Z$ randomly
      9. $\phi^E_{x_1} := \phi^E_{x_1} - \eta \frac{\partial}{\partial \phi^E_{x_1}} (\alpha Z \ell^Z (y, \langle \phi^E_{x_1}, \phi^F_{x_2} \rangle) + \frac{1}{freq_E(x_1)} \Omega(\phi^E_{x_1}))$
      10. $\phi^F_{x_2} := \phi^F_{x_2} - \eta \frac{\partial}{\partial \phi^F_{x_2}} (\alpha Z \ell^Z (y, \langle \phi^E_{x_1}, \phi^F_{x_2} \rangle) + \frac{1}{freq_F(x_2)} \Omega(\phi^F_{x_2}))$
   11. end
   12. end
5. until $||\phi - \phi^{old}|| < \epsilon$
6. return $\phi$
Multi-relational Factorization Model / Evaluation

(a) Ratings

Netflix dataset (sample):
500 users, 3000 movies, 63500 ratings; k=20.
Relations “is_rated” (Netflix) and “has_genre” (IMDB). MAE.

(b) Genres

[Singh and Gordon, 2008b]
Multi-relational Factorization Model / Evaluation

Netflix dataset (sample):
750 users, 1000 movies, 7500 ratings; k=20.
Relations “is_rated” (Netflix) and “has_genre” (IMDB). MAE.

[Singh and Gordon, 2008b]
Multi-relational Factorization Model / Evaluation

Evaluations so far answered most important questions:

1. Joint factorization of auxiliary relations can improve the prediction accuracy of the target relation.
2. Gains are larger for sparse (target) relations.
Multi-relational Factorization Model / Evaluation

Evaluations so far answered most important questions:

1. Joint factorization of auxiliary relations can improve the prediction accuracy of the target relation.
2. Gains are larger for sparse (target) relations.

Open questions:

1. Do multirelational factorization models exploit the auxiliary information better than other models?
   ▶ Evtl. even than models designed with the specific application task in mind?

2. Can the hyperparameters $\alpha$ be identified on the training set?
   ▶ By ex-post picking a value for a hyperparameter that introduces some bias we often can see improvements over the base model.
Example: Coldstart Problem

Target relation (implicit feedback)

Auxiliary relation (social information)

Train users

Test users

Training data:

Test data:
Example: Coldstart Problem / State-of-the-Art Approaches

► First pre-process the auxiliary relation
  ► Dimensionality Reduction
  ► Feature extraction
  ► ...

► Then learn a prediction model on the target relation
Example: Coldstart Problem / Multirelational BPR

\[ \mathcal{E} \] a set called **entity classes**.
\[ X_E \] a set called **entity extensions** (for every \( E \in \mathcal{E} \)).
\[ \mathcal{R} \] a set called (binary) **relations**.
\( s_R \in \mathcal{E}^2 \) the **signature** of relation \( R \in \mathcal{R} \).
\( S_R \subseteq X_E \times X_F \times \mathbb{R} \) a set called **relation extensions** of relation \( R \in \mathcal{R} \),
\( (E, F) := s_R \).
\[ \hat{y}^R(e, f) := \langle \phi^E_e, \phi^F_f \rangle, \quad R \in \mathcal{R}, (E, F) := s_R, (e, f) \in X_E \times X_F \]

regularized loss:

\[
f(\phi; S) := \sum_{R \in \mathcal{R}} \alpha_R \sum_{(e, f, y) \in S_R} \ell^R(y, \hat{y}^R(e, f)) + \sum_{E \in \mathcal{E}} \lambda_E \|\phi^E\|^2
\]

For positive-only relations (\( y \equiv 1 \)):

\[
\sum_{(e, f, y) \in S_R} \ell^R(y, \hat{y}^R(e, f)) \overset{BPR}{\sim} - \sum_{(e, f^+, f^-) \in D_{S_R}} \ln \sigma(\hat{y}^R(e, f^+) - \hat{y}^R(e, f^-))
\]
Example: Coldstart Problem / Learning Multirelational BPR

\[
\text{MR-BPR}(\phi; D) := - \sum_{R \in \mathcal{R}} \alpha_R \sum_{(e, f^+, f^-) \in D_{SR}} \ln \sigma(\hat{y}_R^R(e, f^+) - \hat{y}_R^R(e, f^-)) \\
+ \sum_{E \in \mathcal{E}} \lambda_E \|\phi^E\|^2
\]

1. \text{LearnMR-BPR}(\mathcal{E}, \mathcal{R}, D, \lambda, \alpha, \eta):
2. \text{initialize} \ \hat{\Theta} := (\phi_E)_{E \in \mathcal{E}}
3. \text{repeat}
4. \text{for} \ R \in \mathcal{R} \ \text{do}
5. \quad \text{draw} \ (e, f^+, f^-) \ \text{from} \ D_R
6. \quad \hat{\Theta} \leftarrow \hat{\Theta} + \eta \left( \alpha_R \frac{e^{-\hat{y}_R^R(e, f^+, f^-)}}{1 + e^{-\hat{y}_R^R(e, f^+, f^-)}} \cdot \frac{\partial}{\partial \hat{\Theta}} \hat{y}_R^R(e, f^+, f^-) - \lambda_\Theta \cdot \hat{\Theta} \right)
7. \text{end}
8. \text{until} \ convergenc
9. \text{return} \ \hat{\Theta}
Example: Coldstart Problem / Evaluation

Datasets

<table>
<thead>
<tr>
<th>dataset</th>
<th>relation</th>
<th>dimensionality</th>
<th># instances</th>
<th>avg. instances</th>
<th>sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/user</td>
<td>/label</td>
</tr>
<tr>
<td>Blogcatalog</td>
<td>target</td>
<td>10312x39</td>
<td>14,476</td>
<td>1.4</td>
<td>371.2</td>
</tr>
<tr>
<td></td>
<td>auxiliary</td>
<td>10312²</td>
<td>667,966</td>
<td>64.7</td>
<td>-</td>
</tr>
<tr>
<td>Flickr</td>
<td>target</td>
<td>80513x195</td>
<td>107,741</td>
<td>1.3</td>
<td>552.5</td>
</tr>
<tr>
<td></td>
<td>auxiliary</td>
<td>80513²</td>
<td>11,799,764</td>
<td>146.6</td>
<td>-</td>
</tr>
<tr>
<td>YouTube</td>
<td>target</td>
<td>31,703x47</td>
<td>50,691</td>
<td>1.6</td>
<td>1,078.5</td>
</tr>
<tr>
<td></td>
<td>auxiliary</td>
<td>1, 138, 499²</td>
<td>5,980,886</td>
<td>5.3</td>
<td>-</td>
</tr>
</tbody>
</table>

Evaluated methods

- modularity maximization [Tang and Liu, 2009a]
- edge clustering [Tang and Liu, 2009b]
- BPR-map [Gantner et al., 2010]
Example: Coldstart Problem / Evaluation (Macro-F1)

[Macro-F1 vs. Training-Percentage]

[Krohn-Grimberghe et al., 2012]
Example: Coldstart Problem / Evaluation (AUC)

[Krohn-Grimberghe et al., 2012]
Multi-relational Factorization Models / Summary

- Factorization models can be learned for several relations (with some common modes) at once
  - latent features are associated to modes / entity classes
    - so that the factorizations share latent features
  - the loss of auxiliary relations has to be weighted down
    - (hyperparameters)

- Parameters of multi-relational factorization models should not be learned sequentially, but jointly
  - e.g., interleaving update steps of different relations of iterative learning algorithms such as SGD or ALS.

- Many problems can be modeled by multi-relational factorization problems, e.g., attribute-aware recommender systems, link prediction with node and link features.
Outline

1. Matrix Factorization Models for Binary Relations
2. Learning Matrix Factorization Models
3. Unary and Ordinal Targets and Ranking
4. Multi-Relational Factorization Models
5. Bayesian Matrix Factorization
6. Tensor Factorization Models for Higher Arity Relations
7. Factorization Models Involving Time
8. Factorization Machines
Probabilistic MF

Probabilistic interpretation for MF with L2 loss and L2 regularization:

\[ p(Y \mid \Phi, \Psi, \alpha) := \prod_{(u,i,y) \in Y} \mathcal{N}(y \mid \Phi_u^T \Psi_i, \alpha^{-1}) \]

\[ p(\Phi \mid \alpha_\Phi) := \prod_{u=1}^U \mathcal{N}(\Phi_u \mid 0, \alpha_\Phi^{-1} I) \]

\[ p(\Psi \mid \alpha_\Psi) := \prod_{i=1}^I \mathcal{N}(\Psi_i \mid 0, \alpha_\Psi^{-1} I) \]

Then the negative loglikelihood is just

\[ f(\Phi, \Psi \mid Y, \alpha) := -\log p(Y \mid \Phi, \Psi, \alpha) p(\Phi \mid \alpha_\Phi) p(\Psi \mid \alpha_\Psi) \]

\[ \propto \alpha \sum_{(u,i,y) \in Y} (y - \Phi_u^T \Psi_i)^2 + \alpha_\Phi \sum_{u=1}^U \Phi_u^T \Phi_u + \alpha_\Psi \sum_{i=1}^I \Psi_i^T \Psi_i \]

\[ \propto \sum_{(u,i,y) \in Y} (y - \Phi_u^T \Psi_i)^2 + \lambda_\Phi \|\Phi\|^2 + \lambda_\Psi \|\Psi\|^2, \quad \lambda_\Phi := \frac{\alpha_\Phi}{\alpha}, \lambda_\Psi := \frac{\alpha_\Psi}{\alpha} \]
Bayesian Probabilistic MF

Adding hyperpriors:

\[ p(Y \mid \Phi, \Psi, \alpha) := \prod_{(u,i,y) \in Y} \mathcal{N}(y \mid \Phi_u^T \Psi_i, \alpha^{-1}) \]

\[ p(\Phi \mid \mu_\Phi, \Lambda_\Phi) := \prod_{u=1}^{U} \mathcal{N}(\Phi_u \mid \mu_\Phi, \Lambda_\Phi) \]

\[ p(\Psi \mid \mu_\Psi, \Lambda_\Psi) := \prod_{i=1}^{I} \mathcal{N}(\Psi_i \mid \mu_\Psi, \Lambda_\Psi) \]

\[ p(\mu_\Phi, \Lambda_\Phi \mid \mu_0, \nu_0, \beta_0, W_0) := \mathcal{N}(\mu_\Phi \mid \mu_0, (\beta_0 \Lambda_\Phi)^{-1}) \mathcal{W}(\Lambda_\Phi \mid W_0, \nu_0) \]

\[ p(\mu_\Psi, \Lambda_\Psi \mid \mu_0, \nu_0, \beta_0, W_0) := \mathcal{N}(\mu_\Psi \mid \mu_0, (\beta_0 \Lambda_\Psi)^{-1}) \mathcal{W}(\Lambda_\Psi \mid W_0, \nu_0) \]
Bayesian Probabilistic MF

adapted from Salakhutdinov and Mnih [2008]
Bayesian Probabilistic MF / Inference

As usual, one cannot analytically compute

\[
p(Y_{u,i} \mid Y, \alpha, \mu_0, \nu_0, \beta_0, W_0) := \int p(Y_{u,i} \mid \Phi_u, \Psi_i, \alpha) \, p(\Phi_u, \Psi_i \mid Y, \mu_\Phi, \Lambda_\Phi, \mu_\Psi, \Lambda_\Psi) \\
p(\mu_\Phi, \Lambda_\Phi \mid \mu_0, \nu_0, \beta_0, W_0) \, p(\mu_\Psi, \Lambda_\Psi \mid \mu_0, \nu_0, \beta_0, W_0) \\
d(\Phi_u, \Psi_i, \mu_\Phi, \Lambda_\Phi, \mu_\Psi, \Lambda_\Psi)
\]
Bayesian Probabilistic MF / Gibbs

But one can draw from conditional distributions sequentially (Gibbs):

1. draw new user features:

\[ p(\Phi_u \mid Y, \Psi, \mu_\Phi, \Lambda_\Phi, \alpha) := \mathcal{N}(\Phi_u \mid \mu^*_u, (\Lambda^*_u)^{-1}) \]

with

\[ \Lambda^*_u := \Lambda_\Phi + \alpha \sum_{(u',i,y) \in Y: u' = u} \Psi_i \Psi_i^T \]

\[ \mu^*_u := (\Lambda^*_u)^{-1} (\alpha \sum_{(u',i,y) \in Y: u' = u} \Psi_i y) + \Lambda_\Phi \mu_\Phi \]

2. draw new user feature precision matrices:

\[ p(\Lambda_\Phi \mid \Phi, \mu_0, \nu_0, \beta_0, W_0) := \mathcal{W}(\Lambda_\Phi \mid W_0^*, \nu_0^*) \]

with

\[ (W_0^*)^{-1} := W_0^{-1} + U \bar{S} + \frac{\beta_0 U}{\beta_0 + U} (\mu_0 - \bar{\Phi})(\mu_0 - \bar{\Phi})^T \]

\[ \bar{S} := \frac{1}{U} \sum_{u=1}^{U} \Phi_u \Phi_u^T, \quad \bar{\Phi} := \frac{1}{U} \sum_{u=1}^{U} \Phi_u, \quad \nu_0^* := \nu_0 + U \]
Bayesian Probabilistic MF / Gibbs

But one can draw from conditional distributions sequentially (Gibbs):

3. draw new user feature means:

\[
p(\mu_\Phi \mid \Phi, \Lambda_\Phi, \mu_0, \nu_0, \beta_0, W_0) := \mathcal{N}(\mu_\Phi \mid \mu_0^*, (\beta_0^* \Lambda_\Phi)^{-1})
\]

with \( \mu_0^* := \frac{\beta_0 \mu_0 + U\bar{\Phi}}{\beta_0 + U} \), \( \beta_0^* := \beta_0 + U \),

1b, 2b, 3b. draw new item features \( \Psi_i \), item feature means \( \mu_\Psi \) and item feature precision matrices \( \Lambda_\Psi \)

(same formulas, just swap \( \Phi \) and \( \Psi \) and \( u \) and \( i \))

[Salakhutdinov and Mnih, 2008]
Bayesian Probabilistic MF / Gibbs

1. SGD-BPMF($Y, K, T_0, T, \alpha, \mu_0, \nu_0, \beta_0, W_0$):
2. $(\Phi^1, \Psi^1) := \text{learn-MF}(Y, K)$
3. for $t = 1, \ldots, T_0 + T$ do
4. $\Lambda_t^\Phi \sim p(\Lambda_\Phi | \Phi^t, \mu_0, \nu_0, \beta_0, W_0)$
5. $\mu_t^\Phi \sim p(\mu_\Phi | \Phi^t, \Lambda_t^\Phi, \mu_0, \nu_0, \beta_0, W_0)$
6. $\Lambda_t^\Psi \sim p(\Lambda_\Psi | \Psi^t, \mu_0, \nu_0, \beta_0, W_0)$
7. $\mu_t^\Psi \sim p(\mu_\Psi | \Psi^t, \Lambda_t^\Psi, \mu_0, \nu_0, \beta_0, W_0)$
8. for $u = 1, \ldots, U$ do
9. $\Phi_{u}^{t+1} \sim p(\Phi_u \mid Y, \Psi^t, \mu_t^\Phi, \Lambda_t^\Phi, \alpha)$
10. end
11. for $i = 1, \ldots, I$ do
12. $\Psi_{i}^{t+1} \sim p(\Psi_i \mid Y, \Phi^{t+1}, \mu_t^\Psi, \Lambda_t^\Psi, \alpha)$
13. end
14. end
15. return $(\Phi^t, \Psi^t)_{t=T_0+1, \ldots, T_0+T}$
Bayesian Probabilistic MF / Gibbs

- hyperhyperparameters (used by Salakhutdinov and Mnih [2008]):

\[ \mu_0 = 0, \quad \nu_0 = K, \quad W_0 = I, \quad \alpha = 2, \quad \beta_0 = 1 \]

- prediction:

\[
\hat{y}(u, i; (\Phi_t, \Psi_t)_{t=1,\ldots,T}) := \frac{1}{T} \sum_{t=1}^{T} \langle \Phi^t_u, \Psi^t_i \rangle
\]
Bayesian Probabilistic MF / Evaluation

[Salakhutdinov and Mnih, 2008]
Bayesian Probabilistic MF / Evaluation

<table>
<thead>
<tr>
<th>D</th>
<th>Valid. RMSE PMF</th>
<th>BPMF</th>
<th>Inc.</th>
<th>Test RMSE PMF</th>
<th>BPMF</th>
<th>Inc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>0.9154</td>
<td>0.8994</td>
<td>1.74</td>
<td>0.9188</td>
<td>0.9029</td>
<td>1.73</td>
</tr>
<tr>
<td>40</td>
<td>0.9135</td>
<td>0.8968</td>
<td>1.83</td>
<td>0.9170</td>
<td>0.9002</td>
<td>1.83</td>
</tr>
<tr>
<td>60</td>
<td>0.9150</td>
<td>0.8954</td>
<td>2.14</td>
<td>0.9185</td>
<td>0.8989</td>
<td>2.13</td>
</tr>
<tr>
<td>150</td>
<td>0.9178</td>
<td>0.8931</td>
<td>2.69</td>
<td>0.9211</td>
<td>0.8965</td>
<td>2.67</td>
</tr>
<tr>
<td>300</td>
<td>0.9231</td>
<td>0.8920</td>
<td>3.37</td>
<td>0.9265</td>
<td>0.8954</td>
<td>3.36</td>
</tr>
</tbody>
</table>

[Salakhutdinov and Mnih, 2008]
Bayesian Factorization Models / Summary

- Factorization models with L2 loss and L2 regularization can be interpreted as probabilistic models with normal targets and latent features.

- Priors — mean and precision of latent features — can be added resulting in a hierarchical Bayesian model.

- For conjugate priors (normal mean, Wishart precision), conditional distributions can be analytically computed and parameters can be learned by Gibbs sampling.

- Bayesian models usually provide better performance than a single model instance (comparable to an ensemble of model instances).
End of Part I
References I


References II


References III


References IV


