Motivation

Autoencoders have been proven very successful in various vision and speech problems. Can we do the same for collaborative filtering?

A: Yes

Rating prediction problem

Given a partially observed user-item rating matrix, \( R^{n \times m} \), fill in the missing entries

AutoRec model

For each item, construct (partially observed) vector of ratings \( \mathbf{r}^{(i)} \)

Perform autoencoding on result, where

- Weights are tied across items
- Only observed ratings are used to update model

Prediction:

\[
\hat{R}_{ui} = (h(\mathbf{r}^{(i)}; \theta))_{ui},
\]

\[
h(\mathbf{r}; \theta) = f(V \cdot g(Vr + \mu) + b)
\]

Training objective:

\[
\min_{\theta} \sum_{i=1}^{m} ||\mathbf{r}^{(i)} - h(\mathbf{r}^{(i)}; \theta)||_2^2 + \frac{\lambda}{2} (||W||_F^2 + ||V||_F^2),
\]

Comparisons with existing methods

<table>
<thead>
<tr>
<th>AutoRec</th>
<th>RBM-CF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model type</td>
<td>Discriminative</td>
</tr>
<tr>
<td>Objective</td>
<td>RMSE</td>
</tr>
<tr>
<td>Optimisation</td>
<td>Gradient-based (fast)</td>
</tr>
<tr>
<td>Ratings</td>
<td>Real-valued</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AutoRec</th>
<th>Matrix Factorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Embedding</td>
<td>Users only</td>
</tr>
<tr>
<td>Representation</td>
<td>Non-linear</td>
</tr>
</tbody>
</table>

Experiments

Q: Is user- or item-based modelling better?

A: Item-based is superior.

Q: What are good choices of activations \( f(\cdot) \), \( g(\cdot) \)?

<table>
<thead>
<tr>
<th>( f(\cdot) )</th>
<th>( g(\cdot) )</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sigmoid</td>
<td>Sigmoid</td>
<td>0.836</td>
</tr>
<tr>
<td>Identity</td>
<td>Sigmoid</td>
<td>0.831</td>
</tr>
<tr>
<td>Identity</td>
<td>Identity</td>
<td>0.852</td>
</tr>
</tbody>
</table>

Q: How many hidden units are needed for AutoRec?

A: Good performance with \( \sim 400 \) hidden units

Q: How does AutoRec perform against all baselines?

A: Systematically outperforms state-of-the-art methods

Q: Do deep extensions of AutoRec help?

A: Deep I-AutoRec with three hidden layers (500-200-500) reduced RMSE from 0.831 to 0.827 on ML-1M dataset.

Try it now: https://github.com/mesuvash/NNRec

Future work

Further exploration of deep autoencoders, and applications to implicit feedback datasets.

References