Recommender System and Data Analysis

Boyoung Kim

Seoul National University

March 3, 2017

Outline

1. Collaborative Filtering Using Auto-encoder
   - Introduction
   - AutoRec (Suvash Sedhain, et al., 2015)
   - Itarative method using Auto-encoder
   - Experimental results

2. Recommendation Based On Click Through Rates
   - Introduction
   - Likelihood based approach
   - Experimental results

Outline

1. Collaborative Filtering Using Auto-encoder
   - Introduction
   - AutoRec (Suvash Sedhain, et al., 2015)
   - Itarative method using Auto-encoder
   - Experimental results
2. Recommendation Based On Click Through Rates
   - Introduction
   - Likelihood based approach
   - Experimental results

Introduction

- Some Deep Learning models can be used to model tabular data, such as user's ratings of movies.

Jester 5k dataset

- 5000 users and 100 jokes.
- Rating between -10.00 and 10.00
- Users give the score for a few jokes.

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10	j11	j12
u883	-0.49	-4.27	0.29	1.89	-1.12	0.39	-2.52	-3.01	-0.78	-4.03	1.75	-0.63
u14061	0.49	2.91	-2.62	NA	-0.19	-0.15	-6.75	2.14	NA	-3.20	5.44	-4.03
u6974	0.49	-6.12	-7.04	-6.50	8.06	-6.65	5.78	3.45	-2.86	7.96	-3.35	-4.81
u5081	NA	NA	NA	NA	2.48	5.97	-8.69	-5.97	NA	NA	NA	-0.10
u3328	NA	-2.28	NA	NA	-7.62	-7.04	-2.38	-7.52	NA	-0.58	-3.01	-3.98
u8103	NA	NA	NA	NA	-3.59	NA	4.56	-0.49	NA	NA	NA	NA

Deep Neural Network (Deep learning)

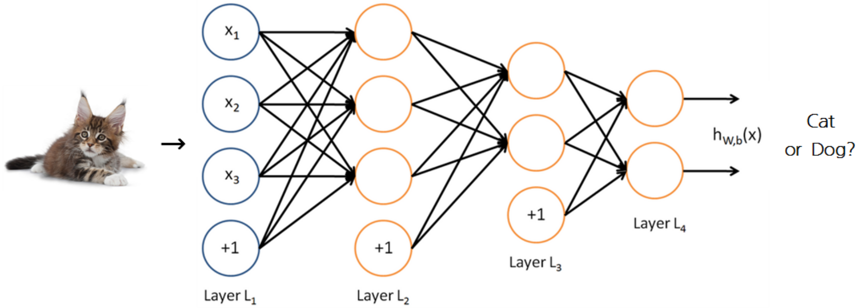

Figure 1: Classification example of deep neural network

Auto-encoder

- Unsupervised learning version of Neural Network.
- AE can be used for dimensionality reduction of high-dimensional data.
- AE generate a hidden representation from an input, and reconstruct the output as the input from the hidden representation.
- Setting the target values to be equal to the input : $\hat{x} \approx x$.

Auto-encoder

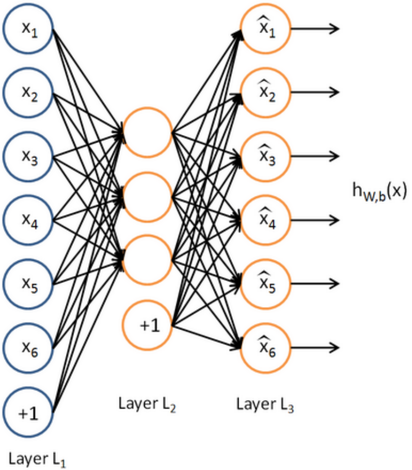

Figure 2: Architecture of autoencoder

Auto-encoder

- Suppose that n inputs and F hidden units.
- Then the hidden and output units are :

$$h_j = g(\sum_{i=1}^{n} V_{ij} x_i + a_j) \quad \text{for } j = 1, \cdots, F \quad \text{and}$$

$$\hat{x}_i = f(\sum_{j=1}^{F} W_{ij} h_j + b_i) \quad \text{for } i = 1, \cdots, n$$

where $\mathbf{a} \in \mathbb{R}^F$ and $\mathbf{b} \in \mathbb{R}^n$ are bias vectors,
$\mathbf{V} \in \mathbb{R}^{n \times F}$ and $\mathbf{W} \in \mathbb{R}^{n \times F}$ are weight matrices and
$f(\cdot)$ and $g(\cdot)$ are activation functions (eg, $f(x) = 1/(1 + e^{-x})$).

AutoRec : Autoencoders Meet Collaborative Filtering (Suvash Sedhain, et al., 2015)

- We use different Autoencoder for each user.
- Item-based AutoRec use Autoencoder for each item.

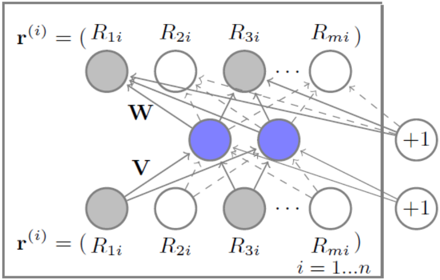

Figure 3: User-based AutoRec model

Itarative method using Auto-encoder

- Fill in 0 or mean of ratings that users have not rated. Consider it as input.
- Get the output from the Auto-encoder model.
- Iterate until convergence
  - Fix the ratings which the users have rated and fill in the predicted values(in the previous step) that users have not rated. Consider it as input.
  - Get the output from the Auto-encoder model.

Experimental results

- Training data : 80%, Test data : 20%
- Test RMSE $= \sqrt{\sum_{u,i \in \text{Test set}} (r_{ui} - \hat{r_{ui}})^2 / |\text{Test set}|}$.

Table 1: Comparision of the test RMSE

Methods	test RMSE
Matrix Factorization	4.1645
Personalized	4.1283
U-Autorec	4.3570
I-Autorec	4.1445
Iterative method	4.2488

Outline

Click Trough Rates data Analysis

- The effective of a particular online ad on a particular website.
- Observe the number of clicks and exposures of the ads to users.
- Predict

$$\textit{Click through rate}(\%) = \frac{\#\ \textit{of clicks}}{\#\ \textit{of exposures}}$$

- Recommend the best ad for a specific user.

Likelihood based approach

- n users, p ads.
- X_{ua} : the number of clicks of the ad a to user u.
- N_{ua} : the number of exposures of the ad a to user u.

Likelihood based approach

- Assume $X_{ua} \sim Bin(N_{ua}, p_{ua})$.
- log likelihood

$$l(\mathbf{p}) = \sum_{u=1}^{n} \sum_{p=1}^{a} \{X_{ua} log p_{ua} + (N_{ua} - X_{ua}) log(1 - p_{ua})\}$$

- Estimate p_{ua} that maximize the log likelihood.

Likelihood based approach

Propose three models for p_{ua}.

1) Ad-wise probability model

- Assume that $p_{ua} = p_a$ for all u.
- ML estimator : $\hat{p_{ua}} = \frac{\sum_u X_{ua}}{\sum_u N_{ua}}$.

2) Additive model

- Assume that *logit* $p_{ua} = \alpha + \beta_u + \gamma_a$.

3) Matrix factorization

- For a pre-selected positive integer K, assume

$$logit\ p_{ua} = \alpha + \beta_u + \gamma_a + \sum_{k=1}^{K} r_{uk} \cdot q_{ka}.$$

Experimental results

Table 2: Predicted probability that user40 clicks the ad205 of models

Model	$\hat{p}_{u=40,a=205}$
Ad-wise	0.00785
Additive	0.00567
Matrix factorization	0.00566

Table 3: Comparision of test log-likelihood of models

Model	Test log-likelihood
Ad-wise	-49489.6
Additive	-47066.9
Matrix factorization	-47033.1