HANDLING INCOMPLETE MATRIX DATA VIA CONTINUOUS-VALUED INFINITE RELATIONAL MODEL

Tomohiko Suzuki, Takuma Nakamura, Yasutoshi Ida and Takashi Matsumoto

Waseda University, Graduate School of Advanced Science and Engineering
3-4-1, Okubo, Shinjuku, Tokyo, 169-8555, Japan
{suzuki08, nakamura10, ida11}@matsumoto.eb.waseda.ac.jp,
takashi@matsumoto.elec.waseda.ac.jp

ABSTRACT

A continuous-valued infinite relational model is proposed as a solution to the co-clustering problem which arises in matrix data or tensor data calculations. The model is a probabilistic model utilizing the framework of Bayesian Nonparametrics which can estimate the number of components in posterior distributions. The original Infinite Relational Model cannot handle continuous-valued or multidimensional data directly. Our proposed model overcomes the data expression restrictions by utilizing the proposed likelihood, which can handle many types of data. The posterior distribution is estimated via variational inference. Using real-world data, we show that the proposed model outperforms the original model in terms of AUC score and efficiency for a movie recommendation task. (111 words)

***Index Terms*—** Machine learning, Bayesian methods, Dirichlet Process, Variational Bayes, Infinite Relational Model

1. INTRODUCTION

Recently, there has been growing interest in the analysis of relational data, such as hyperlinks among webpages, historical lists of items purchased by customers, and so on. Many models have been proposed for this kind of analysis. The stochastic block model (SBM) [1] formulates the co-clustering problem in a probabilistic model. The Infinite Relational Model (IRM) [5] and the Infinite Hidden Relational Model (IHRM) [8] are both extensions of the SBM in a Bayesian Nonparametrics (BNP) [2, 3] framework. The IRM partitions data into clusters, whereas the Mixed Membership Stochastic Block model (MMSB) [6] allows objects belonging to multiple clusters. The Frequency-based Infinite Relational Model (FIRM) [7] accounts for the frequency of the occurrence of a relational data. Dynamic IRM [9] is the time-evolving IRM. These models represent binary states or frequency. In many real-world datasets, however, the relation is not " simply 1 or 0 " or an integer. In addition, although Bayesian Co-Clustering [10] can express any type of data by changing the likelihood, it does not model the presence of data and arbitrary data expression simultaneously. Therefore, we extend the infinite relational models to one that permits the relation representation to be a continuous value or multidimensional data and that simultaneously predicts whether the relation is present or not present. We report a quantitative comparison between the proposed model and the IRM.

2. INFINITE RELATIONAL MODEL

The IRM is a model that can partition objects in relational data. The number of clusters is automatically estimated via Dirichlet Process DP(α,G_0). Here, α is a hyperparameter, and G_0 is the underlying base distribution. We use DP via a stick breaking representation which can be written as follows:

$$G(\theta) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \quad , \quad \theta_k \sim G_0 \tag{1}$$

$$\pi_k = V_k \prod_{l=1}^{k-1} (1 - V_l) \quad , \quad V_k \sim Beta(1, \alpha) \tag{2}$$

where π =(π_1,π_2,π_3,$\cdots$) is a mixing proportion of infinite elements. The stick breaking representation assumes that V_k is drawn from the Beta distribution $Beta(1, \alpha)$. The IRM assumes infinite hidden clusters in "Type1" (Row) and also in "Type2" (Column). For example, let us assume that "Type1" is "User" and "Type2" is "Movie", and that a user i belongs to a user cluster k and a movie j belongs to a movie cluster l. The occurrence of a relation between user i and movie j is parameterized by parameter $\eta_{k,l}$, which stands for the likelihood of the presence of a relation between cluster k in Type1 and cluster l in Type2. The generative model of IRM is as follows:

$$\pi^{T1} | \alpha_1 \sim Stickbreaking(\alpha_1) \tag{3}$$

$$\pi^{T2} | \alpha_2 \sim Stickbreaking(\alpha_2) \tag{4}$$

$$Z_i^{T1} | \pi^{T1} \sim Multinominal(\pi^{T1}) \tag{5}$$

$$Z_j^{T2} | \pi^{T2} \sim Multinominal(\pi^{T2}) \tag{6}$$

$$\eta_{k,l} | b_1, b_2 \sim Beta(b_1, b_2) \tag{7}$$

$$x_{i,j} | Z_i^{T1} = k, Z_j^{T2} = l, \eta_{k,l} \sim Bernoulli(\eta_{k,l}) \tag{8}$$

$$x_{i,j} \in \{0, 1\} (1 \le i \le N_{T1}, 1 \le j \le N_{T2}) \tag{9}$$

$\mathbf{Z} = \{Z_i^{T1}, Z_j^{T2}\}_{i=1,\cdots,N_{T1}, j=1,\cdots,N_{T2}}$, $\theta = \{\eta_{k,l}\}_{k,l=1,\cdots,\infty}$.

Here, N_{T1} is the size of $T1$, N_{T2} is the size of $T2$, k indicates a cluster of $T1$, and l indicates a cluster of $T2$.

In (3) and (4), we sample probability proportions π^{T1} and π^{T2} from the stick breaking representation. Objects in $T1$ are assigned to clusters in proportion to π^{T1} which is drawn from $Sticbreaking(\alpha_1)$, objects in $T2$ are assigned in the same manner. Then, in each submatrix indicated by the combination of cluster k in $T1$ and cluster l in $T2$, in probability $\eta_{k,l}$, $x_{i,j}$ becomes 1, and in $(1 - \eta_{k,l})$, $x_{i,j}$ becomes 0.

978-1-4673-0046-9/12/$26.00 ©2012 IEEE

3. PROPOSED MODEL

3.1. Model formulation

The original IRM does not handle continuous values or multidimensional data directly. In some real-world datasets, there exist various kinds of relations. For example, movie ratings contain ratings given by users, and each user has a user-specific rating bias. Getting rid of such biases is important for user preference analysis of the dataset [12]. Thus, the learning model for the movie-user relational analysis should be able to also treat real numbers, not only binary data. By utilizing the proposed continuous model, we can take user specific biases into account; hence, the clustering performance will be improved.

If we convert the Bernoulli distribution to a normal distribution in (8) in order to use continuous values, the model cannot express the loss of data (in the case of $x = 0$). Instead of this, we assume that data occurrence is drawn from the Bernoulli distribution first, and the continuous value is drawn from the normal distribution second. This is formulated as follows:

$$\pi^{T1}|\alpha_1 \sim Stickbreaking(\alpha_1) \tag{10}$$
$$\pi^{T2}|\alpha_2 \sim Stickbreaking(\alpha_2) \tag{11}$$
$$Z_i^{T1}|\pi^{T1} \sim Multinominal(\pi^{T1}) \tag{12}$$
$$Z_j^{T2}|\pi^{T2} \sim Multinominal(\pi^{T2}) \tag{13}$$
$$\eta_{k,l}|b_1, b_2 \sim Beta(b_1, b_2) \tag{14}$$
$$\Lambda_{k,l}|\Lambda_0, \nu_0 \sim Wishart\left(\Lambda_0, \nu_0\right) \tag{15}$$
$$\mu_{k,l}|m_0, \xi, \Lambda_{k,l} \sim Normal\left(m_0, (\xi\Lambda_{k,l})^{-1}\right) \tag{16}$$
$$x_{i,j}|Z_i^{T1} = k, Z_j^{T2} = l, \eta_{k,l} \sim Bernoulli(\eta_{k,l}) \tag{17}$$

if $(x_{i,j} = 1)$ then draw

$$y_{i,j}|x_{i,j}, \mu_{k,l}, \Lambda_{k,l} \sim Normal\left(\mu_{k,l}, \Lambda_{k,l}^{-1}\right), \tag{18}$$

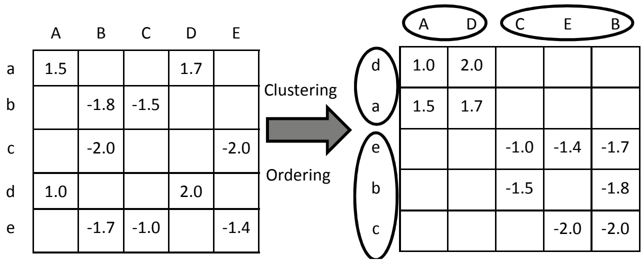
where Λ is a precision matrix, and μ is the mean of the normal distribution. In this construction, the data distribution is not restricted to a normal distribution. It can take the form of an arbitrary function $f(y_{i,j}|\theta_{k,l})$. Our proposed likelihood is explicitly written as:

$$\begin{aligned} &p(y_{i,j}|x_{i,j}, \eta_{k,l}, \theta_{k,l}, Z_i = k, Z_j = l) \\ &= (\eta_{k,l} f(y_{i,j}|\theta_{k,l}))^{I(x_{i,j}=1)}(1-\eta_{k,l})^{I(x_{i,j}=0)} \end{aligned} \tag{19}$$

A schematic illustration of the algorithm is shown in **Fig. 1**.

3.2. Variational inference

We use variational inference instead of Markov Chain Monte Carlo (MCMC) because of the computational cost associated with the latter. The Variational Bayesian inference (VB) iteratively optimizes the variational posterior, which is written in a factorized form $\prod_{k=1}^{K} q(\theta_k|\gamma_k)$, instead of the true posterior $p(\theta|D)$, where γ_k is the variational parameter. Blei and Jordan proposed VB inference for the Dirichlet process [4], where truncated stick breaking (TSB) is used, which is a finite approximation of Setthurman's stick breaking representation. TSB is used for VB inference for both of FIRM [7] and IHRM [11]. The optimization is carried out by maximizing the lower bound of the marginal distribution. Via Jensen's inequality, one has $p(D) \geq E\left[\log \frac{p(D,\Theta)}{q(\Theta)}\right]_{q(\Theta)}$. To obtain the optimal variational posterior $q^*(\theta_k|\gamma_k)$, one takes the variation of the lower bound, and sets it to zero.



	A	B	C	D	E
a	1.5			1.7	
b		-1.8	-1.5		
c		-2.0			-2.0
d	1.0			2.0	
e		-1.7	-1.0		-1.4

	A	D	C	E	B
d	1.0	2.0			
a	1.5	1.7			
e			-1.0	-1.4	-1.7
b			-1.5		-1.8
c				-2.0	-2.0

Fig. 1. Schematic illustration of the proposed algorithm. Upper case characters A, B, C, D, and E represent items, whereas the lower case characters a, b, c, d, and e stand for users. For example, (A, a) with the value 1.5 stands for the fact that user a evaluates item A with score 1.5. Other numerals and characters have similar meanings. The square shown on the left is the input data, whereas the square on the right is the output from the algorithm. Items and users are co-clustered. The proposed model can handle continuous values with incomplete data, in contrast to the original IRM which does not handle continuous values directly.

3.3. VB for proposed model

We assume that the variational posterior has the following factorized form:

$$q(\Theta) = q(Z^{T1})q(Z^{T2})q(V^{T1})q(V^{T2})q(\eta)q(\mu)q(\Lambda) \tag{20}$$

The optimal variational distribution of this model is derived as follows:

$$\log q(Z_i^{T1}) = E\left[\log p(D, \mathbf{Z}, \mathbf{V}, \eta, \mu, \Lambda)\right]_{q(\backslash Z^{T1})} \tag{21}$$
$$\log q(Z_j^{T2}) = E\left[\log p(D, \mathbf{Z}, \mathbf{V}, \eta, \mu, \Lambda)\right]_{q(\backslash Z^{T2})} \tag{22}$$
$$q(V_k^{T1}) = Beta(\gamma_{1,k}^{T1}, \gamma_{2,k}^{T1}) \tag{23}$$
$$q(V_l^{T2}) = Beta(\gamma_{1,l}^{T2}, \gamma_{2,l}^{T2}) \tag{24}$$
$$q(\eta_{k,l}) = Beta(\tau_{1,k,l}, \tau_{2,k,l}) \tag{25}$$

where

$$\gamma_{1,k}^{T1} = 1 + m_k^{T1}, \quad \gamma_{2,k}^{T1} = \alpha_1 + \sum_{k'=k+1}^{K} m_{k'}^{T1} \tag{26}$$

$$\gamma_{1,l}^{T2} = 1 + m_l^{T2}, \quad \gamma_{2,l}^{T2} = \alpha_2 + \sum_{l'=l+1}^{L} m_{l'}^{T1} \tag{27}$$

$$m_k^{T1} = \sum_{i=1}^{N_{T1}} q(Z_i^{N_{T1}}), \quad m_l^{T2} = \sum_{j=1}^{N_{T2}} q(Z_j^{N_{T2}}) \tag{28}$$

$$\tau_{1,k,l} = b_1 + \sum_{i=1}^{N_{T1}} \sum_{j=1}^{N_{T2}} q(Z_i^{T1})q(Z_j^{T2})I(x=1) \tag{29}$$

$$\tau_{2,k,l} = b_2 + \sum_{i=1}^{N_{T1}} \sum_{j=1}^{N_{T2}} q(Z_i^{T1})q(Z_j^{T2})I(x=0). \tag{30}$$

Here, K is the truncation number of infinite components in $T1$, and L is that for $T2$. $E\left[\right]_{q(\backslash Z^{T1})}$ is expectation by $q(\Theta)$ except for $q(Z^{T1})$. Estimation of $q(\mu_{k,l})$,$q(\Lambda_{k,l})$ is similar to that of variational inference for the Gaussian mixture model.

4. EXPERIMENT

4.1. Settings

In the experiments reported below, we used the Movielens dataset, which contains 100,000 ratings from 943 users for 1682 movies. The Movielens dataset is 95% sparse. We made disjoint test sets for five-fold cross validation. For each subset, we ran experiment five times and calculated the average score. We evaluated the clustering result in terms of ”Purity” and ”Coverage” and evaluated the prediction accuracy in terms of the presence or absence via Area Under Curve (AUC) in ROC curves. Since the original IRM cannot express integers and continuous values, we must binarize the ratings. Ratings lower than the means of each user were treated as 0, and the rest were treated as 1. We followed Zhao’s scheme [11] for this. For the continuous IRM, all of the ratings were fully used for cluster analysis, and we subtracted each user’s mean from all of the ratings. We ran VB for the IRM and the proposed model. Except for Normal distribution of the proposed model, we set hyperparameters to the same values. Here, $\alpha_1, \alpha_2 = 10$, and $b_1, b_2 = 0.1$. truncation number K and L is set to 20.

4.2. Evaluation criteria for clustering

Each movie in the Movielens dataset has one or more genre labels, so that we modified the standard definitions of Purity and Coverage as follows.
Purity: We regarded the most-observed genre in a cluster as the cluster label, then computed purity in each cluster, and took an average weighted by the cluster sizes.
Coverage: If one movie genre is the most popular genre in a cluster, we increased the Coverage value by one. When the most popular genre in many clusters was the same, we only increased the Coverage by one. The maximum Coverage was 19 because Movielens has 19 genre labels.
To calculate Purity and Coverage, we need class assignment of each movie. Using the estimated variational posteriors $q(Z_j^{T2})$, we regard the class index l' that maximizes $q(Z_j^{T2} = l')$ as a cluster assignment of movie j.

4.3. Prediction accuracy

Prediction accuracy indicates whether the proposed algorithm correctly predicts the presence of ratings. Predictive density $p(x_{i,j}|D)$ is used for the prediction. First, we calculated the prediction accuracy using a threshold $p(x_{i,j}|D) \geq 0.5$. When $p(x_{i,j}|D)$ was higher than 0.5, we assumed that user i rates movie j. Second, changing the threshold from 0 to 1, we drew an ROC curve and calculated AUC. In an experiment for each cross validation set, 80000 ratings were used for posterior inference, and 20000 rated ratings and 20000 unrated ratings were tested by the predictive distribution.

4.4. Recommendation accuracy

Recommendation accuracy indicates whether the proposed algorithm predicts users’ preference for movies. We assumed that user i prefered movie j if $rating(i,j)$-$userbias(i)$≥0. Here $rating(i,j)$ is the user i’s rating on the movie j, and $usermean(i)$ is the mean of user i’s ratings. For original IRM, we used predictive density $p(x_{i,j}|D)$. Due to the sparseness of the datasets, the problem of choosing thresholds arouse. For the original IRM, We tested 11 thresholds for the predictive distribution. Those are $0.001, 0.01, 0.1, 0.2, 0.3, \ldots, 0.9$. For the proposed model, predictive expectation of $y_{i,j}$ was used. Predictive expectation of $y_{i,j}$ indicates the prediction for the user i’s rating for the movie j. If predictive expectation of $y_{i,j}$ was higher than 0, now that we subtracted user-specific biases from ratings, we assumed user i prefered movie j. We used 5 fold cross validation set. In a experiment, 80,000 ratings were used to estimate the posterior distribution, and 20,000 ratings were tested.

4.5. Results

The proposed model outperformed the original IRM especially in terms of AUC of ROC curves and Coverage, as shown in **Table 1** and **Fig. 2**. In the AUC comparison, our proposed model improved the AUC score of the original IRM from 0.807 to 0.845. Because of the sparseness of the Movielens dataset, the probability of presence tends to be small. To obtain better prediction accuracy in real-world data, we should set the threshold to one suitable for the data density. Obtained AUC score suggests us to set the threshold to appropriate one. **Fig. 3** shows a graph of the number of movies in the clusters. Observe that the number of movies in the first cluster decreased from 676 to 444 with the proposed algorithm. The number of movies in the second cluster also decreased from 202 to 173. In the original IRM, the largest cluster contained more than 600 movies, which amounts to almost 1/3 of all movies. Many of the members in this cluster should be in different clusters. This phenomenon is caused by the binarization, which collapses 20% of all the ratings; hence, those movies that are evaluated only a small number of times tend to be included in this cluster. In contrast, in the continuous IRM proposed here, such a phenomenon is avoided. **Fig. 4** shows a graph of Recommendation accuracy. For the original IRM, the highest mean score was 0.600±0.022 (mean%± standard deviation) in the case that the threshold equals to 0.1. Because of the sparseness of the datasets, it is difficult to single out an appropriate threshold for the original IRM whereas the proposed model got higher Recommendation accuracy 0.613±0.010 in average. Our proposed model enjoyed the benefits of using predictive expectation for ratings. This was oweing to the generative model which generates continuous-valued data.

Table 1. Performance comparison. mean ± standard deviation.

	Continuous IRM	Original IRM
Prediction accuracy	0.560±0.009	0.523±0.007
AUC	0.845±0.044	0.807±0.015
Coverage	4.600±1.000	3.240±0.523
Purity	0.437±0.004	0.434±0.003

5. DISCUSSION

We extended the original IRM to a model that can handle continuous values and many types of data. In the Movielens dataset, we compared prediction accuracies and clustering performance between the original IRM and the proposed method. The proposed model expresses the presence of data and the continuous value simultaneously, and enables us to use not only all the ratings in the learning sets but also the unbiased continuous ratings. The proposed model produced improved results especially in ”AUC score” and ”the efficiency in the recommendation task”. Our model can easily treat richer data representations, such as relations with multi-dimensional vectors or tree-structured Bayesian nets, or multinominal document

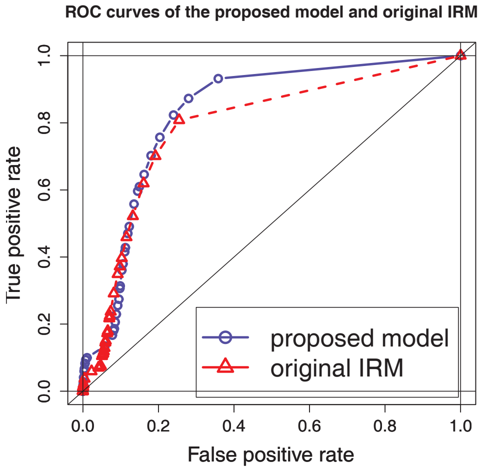


Fig. 2. Comparison of ROC curves between the proposed model and the original IRM for a subset of cross validation sets. Whether a rating is present or not present is predicted using the predictive distribution. The ROC is calculated for 20000 rated ratings and another 20000 unrated ratings, by changing the threshold for the prediction.

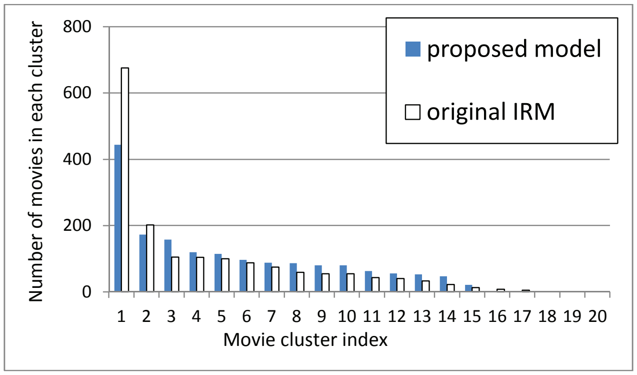


Fig. 3. Clustering result of movies in Movielens dataset using the proposed model.

models, if the likelihood function is modified in an appropriate manner. The idea is also applicable to dynamic models, which is one of our future projects.

6. ACKNOWLEDGEMENT

The authors thank to Kenichi Kurihara for giving us the source code of FIRM. The implementation helped us to make our experiments faster.

7. REFERENCES

[1] K. Nowicki and T. Snijders, "Estimation and prediction for stochastic blockstructures," *JASA,* Vol. 96 No. 455, pp. 1077-1087, 2001

[2] T. Ferguson, "A Bayesian Analysis of Some Nonparametric Problems," *Annals of Statistics,* Vol. 1, No. 2, pp. 209-230, 1973

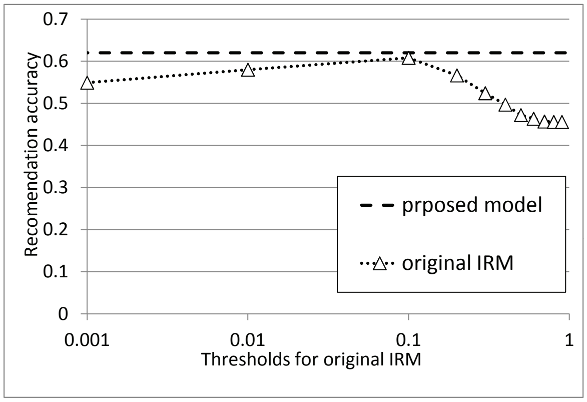


Fig. 4. Comparison of Recommendation accuracy between the prposed model and the original IRM. Because of the sparseness of the datasets, the original IRM must choose an appropriate threshold for the predictive distribution $p(x_{i,j}|D)$(distribution for presence of high ratings) to estimate user preference, whereas the proposed model is utilizing predictive mean of $y_{i,j}$ (ratings which are subtracted user-specific biases), and does not have to choose a such threshold.

[3] J. Sethuraman, "A Constructive Definition of Dirichlet Priors," *Statistica Sinica,* Vol. 4, pp. 639-650, 1994

[4] D. M. Blei and M. I. Jordan, "Variational Inference for Dirichlet Process Mixtures," *Bayesian Analysis,* Vol. 1, No. 1, pp. 121-144, 2006

[5] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada and N. Ueda, "Learning Systems of Concepts with an Infinite Relational Model," *National Conference on Artificial Intelligence,* 2006

[6] E. Airoldi, D. Blei, S. Fienberg, and E. Xing. "Mixed Membership Stochastic Blockmodels," *Technical report,* arXiv:0705.4485v1 [stat.ME], 2007.

[7] K. Kurihara, Y. Kameya, and T. Sato, "Discovering Concepts from Word Co-occurrences with a Relational Model," *Transactions of the Japanese Society for Artificial Intelligence,* Vol. 22, No. 2, pp. 218-226, 2007

[8] Z. Xu, V. Tresp, K. Yu, and H. Kriegel, "Infinite Hidden Relational Models," *In Proceedings of the 22nd International Conference on Uncertainity in Artificial Intelligence,* 2006

[9] K. Ishiguro, T. Iwata, N. Ueda, and J. Tenenbaum, "Dynamic Infinite Relational Model for Time-varying Relational Data Analysis," *Advances in Neural Information Processing Systems 23,* pp. 919-927, 2010

[10] H. Shan and A. Banerjee, "Bayesian Co-clustering," *Proceedings of the 8th IEEE International Conference on Data Mining,* IEEE Computer Society, pp. 530-539, 2008

[11] Z. Xu and V. Tresp, "Statistical relational learning with nonparametric Bayesian models," *Ph.D. Thesis,* University of Munich, 2007

[12] Y. Koren, R. Bell, and C. Volinsky, "MATRIX FACTORIZATION TECHNIQUES FOR RECOMMENDER SYSTEMS," *IEEE Computer,* Volume 42, Issue 8, pp.30-37, 2009