16

Making Personalized Recommendations in E-Commerce

Mu Zhu

University of Waterloo, Waterloo, ON

Many practical problems in our Internet Age can benefit from ideas in statistics. In this chapter, I briefly tell the story of how two statistical ideas can be applied quite naturally to one such problem.

16.1 Introduction

The problem that I will focus on is that of making personalized recommendations in e-commerce. We encounter personalized recommendations everywhere. If you buy a book or watch a movie online, e.g., from `www.amazon.com` or `www.netflix.com`, their recommender systems will suggest a few other books or movies that they "think" you might also like. Table 16.1 contains a hypothetical example, created to illustrate the main ideas. It shows how four different customers would have rated four different books on the scale of 0–100 (see Remark 16.1). In reality, at any given time, the customers will only have revealed their preferences on a limited number of books. For example, they may have purchased a few and explicitly rated a few others. Therefore, we will pretend that only some of the ratings in Table 16.1 are available, while others — in particular, those marked by "?" — are missing, and the recommender system must predict them based on the observed entries. If the predicted rating is high, a recommendation can then be made. In what follows, we will refer more generally to "users" and "items" rather than just "customers" and "books."

The two statistical ideas that I have alluded to are: regression and shrinkage. Linear regression is one of the most widely used statistical techniques. The idea has been around since at least the early 1800s, and some of its early champions included such mathematical giants as Carl Friedrich Gauß. It postulates that a target variable of interest, y, is a linear function of a number of

TABLE 16.1: Illustrative example: Four users rate four books. *Moby Dick* refers to the novel *Moby Dick; or, The Whale* by Melville (1851). *Dreams* refers to the book *The Interpretation of Dreams* by Freud (1913). *Species* refers to the book *The Origins of Species* by Darwin (1859). *Relativity* refers to the book *Relativity* by Einstein (1916). A question mark (?) indicates that the corresponding rating is treated by various methods as if it were missing/unobserved and to be predicted. Predictions closer to these entries here are deemed more accurate.

	Moby Dick	*Dreams*	*Species*	*Relativity*
Alice	90	~~70~~ ?	30	10
Bob	90	70	~~30~~ ?	10
Cathy	10	~~30~~ ?	70	90
David	10	30	~~70~~ ?	90

other variables $x_1, \ldots, x_d$ plus some random noise ϵ,

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_d x_d + \epsilon,$$

and uses data to estimate the linear function — in particular, the coefficients in front of each x_j, $j \in \{1, \ldots, d\}$. Shrinkage estimation (James and Stein, 1961) is a much newer idea, but it has been central to modern statistical practice. For example, while a "natural" way to predict the batting averages of baseball players in a new season is to base the predictions on each player's historical average (y_i), a much better way (see, e.g., Efron and Morris, 1977) is to shrink each y_i toward the overall mean of all players, i.e.,

$$\bar{y} = \frac{1}{n}(y_1 + \cdots + y_n).$$

Remark 16.1. In practice, we will not usually have ratings of such fine resolution. In many cases, we can only obtain rough indications of whether a customer likes or dislikes a certain book, e.g., a binary indicator of whether the customer has purchased it or read a few of its reviews. Explicit ratings are possible, but they rarely go beyond a five-point scale such as "terrible," "bad," "fair," "good," and "excellent." However, I have chosen the finer-than-usual scale deliberately, so that I can demonstrate more easily the differences of various methods. On such a small example (4 customers × 4 books), the predictions made by various methods would differ by too little if I only used a rough scale.

16.2 Nearest Neighbors

There are many different ways to predict the missing entries in Table 16.1; see, for example, a recent review by Feuerverger et al. (2012) and references therein. The nearest neighbors approach (see, e.g., Koren, 2008) is perhaps the most intuitive. For example, to predict Alice's rating (r_A) of Freud's *The Interpretation of Dreams*, we can consider observed ratings of this book — in this case, those given by Bob ($r_B = 70$) and by David ($r_D = 30$) — and ask: whose preferences, Bob's or David's, do we expect to be closer to those of Alice's? Suppose $s(A, B)$ and $s(A, D)$ measure the similarities between Alice and Bob, and between Alice and David, respectively. Then, we can predict r_A to be

$$\left\{\frac{s(A,B)}{s(A,B)+s(A,D)}\right\} \times r_B + \left\{\frac{s(A,D)}{s(A,B)+s(A,D)}\right\} \times r_D,$$

a weighted average of Bob's and David's ratings, each weighted by their respective similarities to Alice. The similarity between two users can be inferred from items that they have already rated in common. In this case, Alice and Bob have both rated Melville's *Moby Dick* and Einstein's *Relativity*, and their ratings of these items are highly similar. On the other hand, Alice and David have both rated the same two items as well, but their ratings of these items are much less similar. It appears, therefore, that $s(A, B) > s(A, D)$. To give a concrete numeric example, suppose the similarity measure $s(\cdot, \cdot)$ were specified in such a way that $s(A, B) = .75 > .25 = s(A, D)$. Then, our prediction of r_A would be

$$\left(\frac{.75}{.75+.25}\right) \times 70 + \left(\frac{.25}{.75+.25}\right) \times 30 = 60.$$

The choice of the similarity measure, $s(\cdot, \cdot)$, clearly plays a major role in this approach, but I will not go into the mathematical details here.

16.3 Matrix Factorization

A slightly more abstract approach — the matrix factorization approach (see, e.g., Koren et al., 2009) — became popular as a result of the highly publicized Netflix Prize (`www.netflixprize.com`). The data in Table 16.1 form a user-item rating matrix, R, typically with many missing entries. In general, suppose there are n users and m items ($n = m = 4$ in Table 16.1). After accounting for both user-effects and item-effects (more on this below), the matrix factorization approach aims to factor the matrix R into the product of two low-rank matrices,

$$R \approx PQ^\top = \underbrace{\begin{pmatrix} p_1^\top \\ \vdots \\ p_n^\top \end{pmatrix}}_{n \times K} \underbrace{\begin{pmatrix} q_1 & \dots & q_m \end{pmatrix}}_{K \times m}, \tag{16.1}$$

where p_u, q_i are vectors in $\mathbb{R}^K$ and $K \ll \min(n, m)$. The user- and item-effects refer to the fact that some users are more difficult to please, while some items are better liked in general than others. After removing the overall mean of R, these effects can be estimated by the row-means and column-means of R, and are typically removed as well prior to performing the factorization (16.1). In other words, the matrix factorization approach aims to estimate latent coordinates, $p_u, q_i \in \mathbb{R}^K$, respectively for each user (u) and for each item (i), such that the user-item rating (r_{ui}) can be approximated — modulo user- and item-effects — by $p_u^\top q_i$, a measure of how closely aligned the user- and item-coordinates are.

If the coordinates are two-dimensional (i.e., $K = 2$), then this process will literally give us a map of all users and items (Figure 16.1, I). In order to make recommendations to a user, all we have to do is to first locate the user on the map, and then recommend items that are close by (see Remark 16.2). For $K > 2$, the idea is exactly the same, except that the map is high-dimensional. In practice, the parameter K is determined empirically, but generally should be chosen so that the total number of parameters being estimated ($nK + mK$) is considerably smaller than the total number of observed ratings. The key, of course, lies in our ability to create such a user-item map. This can be accomplished by solving a regularized optimization problem, but I will, again, omit the mathematical details.

Remark 16.2. One may notice that, in Figure 16.1 (I), Bob is much farther away from *Moby Dick* than Alice is, even though they both have given it the same rating of 90 (Table 16.1). This is because user- and item-effects have been removed prior to matrix factorization. Based on available ratings, Bob appears to be less critical than Alice is — in particular, Bob's average rating is $(90+70+10)/3 \approx 56.67$, whereas Alice's average rating is $(90+30+10)/3 \approx 43.33$. That's why a book has to be much closer to Alice for her to give it a high rating, but it doesn't have to be as close to Bob for him to give it an equally high rating. The same explanation applies to Cathy and David.

16.4 Matrix Completion

Lately, an even more abstract approach — the matrix completion approach — has attracted some attention as well. The idea is to fill in the missing

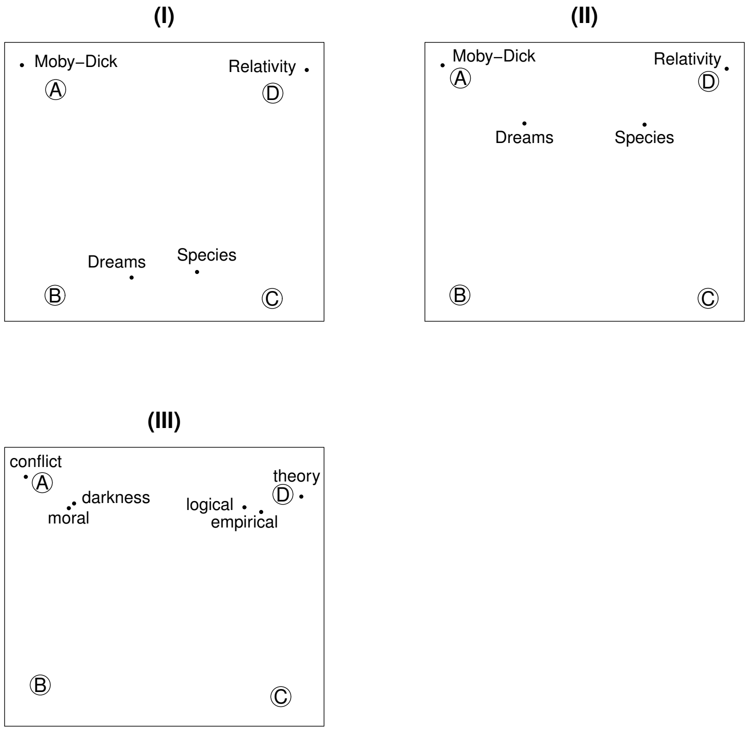

FIGURE 16.1: Illustrative example: Maps produced by different matrix factorization methods, after having removed user- and item-effects. A = Alice; B = Bob; C = Cathy; D = David. (I) A map of users and items, without incorporating any content information. (II) A map of users and items, incorporating content information by the shrinkage approach. (III) A map of users and items' content features, incorporating content information by the regression approach.

entries of R in such a way that the completed matrix — call it $\widehat{R}$ — is as low-rank as possible. Mathematically, this amounts to solving a constrained rank minimization problem (Candès and Recht, 2009). The rationale behind rank minimization is similar to that behind the matrix factorization approach: we believe that user-preferences are driven by only a few key factors; therefore, the rank of the rating matrix cannot be very high. Thus, the two approaches — matrix factorization and matrix completion — share a common philosophical underpinning, but they differ in that the matrix factorization approach is more explicit about the nature of the low-rankness. Rank minimization by itself is an NP-hard problem, essentially meaning that there is currently no way to guarantee an exact solution except when the size of the problem is very small. However, recent theoretical advances (e.g., Candès and Tao, 2005; Candès and Recht, 2009; Recht et al., 2010) have established that, under certain conditions, we can obtain the same solution by solving a much easier, convex optimization problem instead, replacing $\text{rank}(\widehat{R})$ with $\|\widehat{R}\|_*$, the so-called "nuclear norm" of $\widehat{R}$ (see Remark 16.3). The mathematical details here are very technical, and we will definitely stay away from them in this chapter.

Remark 16.3. For those with enough background to appreciate why the so-called ℓ_1-norm has been so important for high-dimensional problems in statistics (see, e.g., Tibshirani, 1996; Donoho, 2006), the nuclear norm of $\widehat{R}$ is defined as

$$\|\widehat{R}\|_* = \sum_k |\sigma_k|^1,$$

where $\sigma_1, \sigma_2, \ldots$ are the singular values of $\widehat{R}$. Recall that $\text{rank}(\widehat{R}) = \sum_k |\sigma_k|^0$. Hence, if we write $\mathbf{v} = (\sigma_1, \sigma_2, \ldots)^\top$ as the vector stacking all the singular values together, then $\text{rank}(\widehat{R})$ and $\|\widehat{R}\|_*$ are just the ℓ_0- and ℓ_1-norms of $\mathbf{v}$, respectively. Therefore, nuclear-norm minimization is to rank minimization what the lasso (Tibshirani, 1996) is to subset selection. For additional information about the lasso, see Chapter 5 by Rob Tibshirani.

16.5 Content-Boosted Matrix Factorization

Sometimes, we may have additional content information about the items. For example, Table 16.2 contains some features that can be used to describe the four books listed in Table 16.1. According to this table, Melville's *Moby Dick* shares three features in common with Freud's *The Interpretation of Dreams*, but only one with Darwin's *The Origin of Species*. Clearly, such information may explain why some users prefer certain items to others. In our illustrative example, for instance, the ratings are highly predictable from the items' content features — any user's ratings of any two items are always $20 \times (4 - Z)$ points apart if the two items share Z content features in common ($Z = 0, 1, 2$, or 3). In reality, the content features will not have such a strong bearing on the users' ratings, but they are still more likely than not to be at least partially informative. If so, they can (and should) be exploited to enhance the predictions of the recommender system. We have proposed two different ways (Forbes and Zhu, 2011; Nguyen and Zhu, 2013) to incorporate such content information into the matrix factorization approach, which we discussed two paragraphs ago. Our proposals are natural applications of the two statistical ideas that I mentioned at the beginning of this chapter.

For example, we can bias two items' coordinates to be close to each other if they share at least a certain number of common features (Figure 16.1, II).

TABLE 16.2: Illustrative example: Content information about the four books in Table 16.1. Italicized words are used as abbreviated descriptions of each feature in Figure 16.1, III.

	Moby Dick	*Dreams*	*Species*	*Relativity*
Themes of *conflict*	√	√	√	×
Elements of *moral* philosophy	√	√	×	×
Darkness of human nature	√	√	×	×
Other *empirical* evidence	×	×	√	√
Logical rigor	×	×	√	√
A grand new *theory*	×	√	√	√

Suppose that each item i is associated with a binary content vector a_i (e.g., a column in Table 16.2, taking "√" as 1 and "×" as 0). Then,

$$\mathcal{S}_c(i) = \{i' : \quad a_i^\top a_{i'} \geq c\}$$

is the set of all items that share at least c common features with i. In the iterative procedure to estimate (p_u, q_i), this added bias amounts to *shrinking* the coordinates of each item (q_i) at every step toward the mean coordinates of those that share enough common features with it — that is, shrinking q_i toward

$$\sum_{i' \in \mathcal{S}_c(i)} \frac{q_{i'}}{|\mathcal{S}_c(i)|},$$

where $|\mathcal{S}_c(i)|$ denotes the size of the set $\mathcal{S}_c(i)$. We call this the shrinkage approach.

Alternatively, we can force an item's coordinates to depend on the item's content features by means of a *regression* relationship, i.e.,

$$q_i = Ba_i. \tag{16.2}$$

In a K-dimensional user-item map, each item i has K coordinates. For each item i, Equation (16.2) actually encodes K simultaneous regression relations,

$$q_i(k) = \sum_j B(k,j)a_i(j), \tag{16.3}$$

one for each coordinate k. Under the constraint (16.2), the problem becomes one of estimating (p_u, B) rather than (p_u, q_i). This can be accomplished by making a relatively small change to the iterative procedure for estimating (p_u, q_i). We call this the regression approach.

For our illustrative example (Tables 16.1–16.2), predictions made by these different matrix factorization approaches (all using $K = 2$) are given in Table 16.3. By directly comparing with Table 16.1, we can see that incorporating content information contained in Table 16.2, whether by shrinkage or by regression, has indeed led to more accurate predictions of the four "missing" entries (emboldened in Table 16.3).

TABLE 16.3: Illustrative example: Predictions made by different matrix factorization methods using $K = 2$. (I) Without incorporating any content information. (II) Incorporating content information by the shrinkage approach. (III) Incorporating content information by the regression approach.

		Moby Dick	*Dreams*	*Species*	*Relativity*
(I)	Alice	89	**48**	30	10
	Bob	90	70	**51**	11
	Cathy	11	**52**	70	90
	David	10	30	**49**	89
(II)	Alice	89	**63**	30	10
	Bob	90	70	**37**	11
	Cathy	11	**37**	70	90
	David	10	30	**63**	89
(III)	Alice	90	**75**	30	10
	Bob	90	70	**25**	10
	Cathy	10	**25**	70	90
	David	10	30	**75**	90

An interesting by-product of the regression approach is that each column of the matrix B — a vector in $\mathbb{R}^K$ — can be interpreted as the latent coordinates for each corresponding content feature. To see this, notice that Equation (16.3) can be interpreted as

$$(k\text{th coordinate of item } i) = \sum_j (k\text{th coordinate of feature } j) \times \mathbf{1}(\text{item } i \text{ has feature } j),$$

where $\mathbf{1}(E)$ is an indicator function taking on the values of 1 or 0 depending on whether E is true or false. Therefore, not only can we create a user-item map to facilitate personalized recommendations, we can also put content features onto the same map (Figure 16.1, III), and gain fresh insight about the content features themselves. For example, using data from `http://allrecipes.com/` and treating ingredients as content features of recipes, we have found "mozzarella" and "firm tofu" to be similar ingredients, but "cottage cheese" and "Swiss cheese" to be dissimilar ones; using the "MovieLens 100K" data from `http://www.grouplens.org/` and treating genres as content features of movies, we have found that "action" movies are more similar to "science fiction" movies than to "war" movies, whereas "war" movies are closer to "animation" movies than to "action" movies (Nguyen and Zhu, 2013).

16.6 Discussion

We are currently contemplating how to incorporate content information into the matrix completion approach. As mentioned earlier, the matrix completion approach is based on the premise that the completed matrix should be low-rank, without being explicit about the nature of the low-rankness. The lack of an explicit parameterization makes the kind of extensions we have proposed to the matrix factorization approach elusive, and it appears to us that a different paradigm is needed altogether. There are certainly many opportunities for statisticians to make contributions.

Acknowledgments

The author's research is supported by the Natural Sciences and Engineering Research Council of Canada. Although the illustrative example (Tables 16.1–16.2) has been created by the author specifically for this chapter, the research being highlighted is based on joint work with his former undergraduate students: Peter Forbes, now a doctoral student at the University of Oxford, and Jennifer Nguyen, now a master's student at University College London. The author thanks the editor for inviting him to contribute this chapter to the special volume. He also thanks the editor, an associate editor, and two anonymous reviewers for their comments and suggestions.

About the Author

Mu Zhu is an associate professor of statistics at the University of Waterloo. A *Phi Beta Kappa* graduate of Harvard University, he obtained his PhD from Stanford University. His primary research interests are machine learning, multivariate analysis, and health care informatics. In 2012–13, he served as president of the Business and Industrial Statistics Section of the Statistical Society of Canada.

Bibliography

Candès, E. J. and Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9:717–772.

Candès, E. J. and Tao, T. (2005). Decoding by linear programming. *IEEE Transactions on Information Theory*, 51:4203–4215.

Donoho, D. L. (2006). For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59:797–829.

Efron, B. and Morris, C. (1977). Stein's paradox in statistics. *Scientific American*, 236:119–127.

Feuerverger, A., He, Y., and Khatri, S. (2012). Statistical significance of the Netflix challenge. *Statistical Science*, 27:202–231.

Forbes, P. and Zhu, M. (2011). Content-boosted matrix factorization for recommender systems: Experiments with recipe recommendation. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys'11, pp. 261–264, New York.

James, W. and Stein, C. M. (1961). Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Vol. I*, pp. 361–379. University California Press, Berkeley, CA.

Koren, Y. (2008). Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'08, pp. 426–434, ACM, New York.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42:30–37.

Nguyen, J. and Zhu, M. (2013). Content-boosted matrix factorization techniques for recommender systems. *Statistical Analysis and Data Mining*, 6:286–301.

Recht, B., Fazel, M., and Parrilo, P. A. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52:471–501.

Tibshirani, R. J. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288.