ISSN: 2250-3676

[IJESAT] [International Journal of Engineering Science & Advanced Technology] Volume-4, Issue-3, 307-310

Singular Value Decomposition and Applications

JajimoggaRaghavendar
Assoc Professor
NallaMalla Reddy Engineering College
Telangana India

Abstract

The multilayer perceptron, when working in auto-association mode, is sometimes considered as an interesting candidate to perform data compression or dimensionality reduction of the feature space in information processing applications. The present paper shows that, for auto-association, the nonlinearities of the hidden units are useless and that the optimal parameter values can be derived directly by purely linear techniques relying on singular value decomposition and low rank matrix approximation, similar in spirit to the well-known Karhunen-Loève transform. This approach appears thus as an efficient alternative to the general error back-propagation algorithm commonly used for training multilayer perceptrons. Moreover, it also gives a clear interpretation of the rôle of the different parameters.

Introduction

Recommender systems apply data analysis techniques to the problem of helping customers to find which products they would like to purchase especially on the internet. These systems are rapidly becoming a crucial tool in E-commerce on the Web. The tremendous growth of customers and products poses two main challenges for recommender systems. The first challenge is to improve the quality of the recommendations for the customers. Making good recommendations increases the customers desire to purchase products, whereas making bad recommendations may result losing customers. Another challenge is to improve the scalability of the recommendation algorithms. These algorithms are able to respond tens of millions of recommendation requests in real-time. In order to make a system scalable, the response time for the requests should be reduced. However, if the algorithm spends less time for recommendation, the quality of the recommendation decreases. Actually, from this perspective these two challenges are in conflict. For this reason, it is important**to consider both of them simultaneously for the proposed solutions.**

The singular value decomposition of a matrix has many applications. Here I'll focus on an introduction to singular value decomposition and an application in clustering articles by topic. In another notebook (link) I show how singular value decomposition can be used in image compression.

Any matrix A can be decomposed to three matrices U, Σ, and V such that $A = U \Sigma V$, this is called singular value decomposition. The columns of U and V are orthonormal and Σ is diagonal. Most scientific computing packages have a function to compute the singular value decomposition, I won't go into the details of how to find U, Σ and V here. Some sources write the decomposition as $A = U \Sigma V^T$, so that their V^T is our V. The usage in this notebook is consistent with how numpy's singular value decomposition function returns

Although the singular value decomposition has interesting properties from a linear algebra standpoint, I'm going to focus here on some of its applications and skip the derivation and geometric interpretations.

Let \(A\) be a \(m \times n\) matrix with column vectors \(\vec{a}_1, \vec{a}_2, ..., \vec{a}_n\). In the singular value decomposition of \(A\), \(U\) will be \(m \times m\), \(\Sigma\) will be \(m \times n\) and \(V\) will be \(n \times n\). We denote the column vectors of \(U\) as \(\vec{u}_1, \vec{u}_2, ..., \vec{u}_m\) and \(V\) as \(\vec{v}_1, \vec{v}_2, ..., \vec{v}_n\), similarly to \(A\). We'll call the values along the diagonal of \(\Sigma\) as \(\sigma_1, \sigma_2, ...\).

Brief discussion of dimensionality

This section isn't required to understand how singular value decomposition is useful, but I've included it for completeness.

If \(A\) is \(m \times n\) (\(m\) rows and \(n\) columns), \(U\) will be \(m \times m\), \(\Sigma\) will be \(m \times n\) and \(V\) will be \(n \times n\). However, there are only \(r = rank(A)\) non-zero values in \(\Sigma\), i.e. \(\sigma_1, ..., \sigma_r \neq 0\); \(\sigma_{r+1}, ..., \sigma_n = 0\). Therefore columns of \(U\) beyond the \(r^{th}\) column and rows of \(V\) beyond the \(r^{th}\) row do not contribute to \(A\) and are usually omitted, leaving \(U\) an \(m \times r\) matrix, \(\Sigma\) an \(r \times r\) diagonal matrix and \(V\) an \(r \times n\) matrix.

Example with data:

Singular value decomposition can be used to classify similar objects (for example, news articles on a particular topic). Note above that similar \(\vec{a_i}\)'s will have similar \(\vec{v_i}\)'s.

Imagine four blog posts, two about skiing and two about hockey. I've made up some data about five different words and the number of times they appear in each post:

```
importpandasaspd
```

```
c_names= ['post1', 'post2', 'post3',
'post4']
words = ['ice', 'snow', 'tahoe', 'goal',
'puck']
post_words=pd.DataFrame([[4, 4, 6, 2],
                          [6, 1, 0, 5],
                          [3, 0, 0, 5],
                          [0, 6, 5, 1],
                          [0, 4, 5, 0]],
index= words,
columns=c_names)
post_words.index.names= ['word:']
```

methodologies to analyze data more accurately with high performance. Recommender systems are one of the most popular and widespread data analysis tools. A recommender system applies knowledge discovery techniques to the existing data and makes personalized product recommendations during live customer interaction. However, the huge growth of customers and products especially on the internet, poses some challenges for recommender systems, producing high quality recommendations and performing millions of recommendations per second.based on dimension reduction is one of these methods which produces high quality recommendations, but has to undergo very expensive matrix calculations. In this thesis, we propose and experimentally validate some contributions to SVD technique which are based on the user and the item categorization. Besides, we adopt tags to classical 2D (User-Item) SVD technique and report the results of experiments. Results are promising to make more accurate and scalable recommender systems

Son yıllardameydanagelenteknolojikgeliĢmelersonucu, verianaliziĢirketlervearaĢtırmacılaraçısından son dereceönemlibiralanhalinegelmektedir. ġirketler, müĢterileriileilgiliellerindevarolanbilgilerianalizeder ekveilerikikararlarınıalırkenbuanalizlerinegörehareket ederekkarlarınıartırmayaçalıĢmaktadır. ġirketlerinbuihtiyaçlarınaparalelolarak, araĢtırmacılarveriyidahadoğruvehızlıiĢleyebilmekiçi nfarklımetodolojilergeliĢtirmektedir.

Önerisistemleribuaçılardan en popülerve en yaygınverianalizyazılımlarıdır.Birönerisistemimevcut veriyebilgiiĢlemeteknikleriniuygulayıpanalizederek, kullanıcılarınakisiselleĢtirilmiĢürünönerilerisunar.AncaközellikleinternetinyaygınlaĢmasıilemüĢteriveürünsayısındakibüyükartıĢlarönerisistemleriiçinbazıproblemleriberaberindegetirmektedir.Bu problemleryüksekkalitedeönerilerinyapılmasıvesaniyedemilyonlarcaöneriisteğinecevapverebilmektir.Önerisistemlerininperformansınıartırmayayönelik, araĢtırmacılartarafındanbirçokmetodönerilmektedir.BoyutindirgemeyedayalıTekilDeğerAyrıĢımı (TDA), son dereceyüksekkalitedeönerilerüretenfakathesaplamaaçısındanpahalımatrisiĢlemlerigerektirenbiryöntemdir.Bu tezkapsamındaTDA'yadayalıöneritekneğine, kullanıcıyaveürünleregörekategorileroluĢturma, vebukategorileriönerisürecinedahiletmeiĢlemiönerilmektevebueklemeninolumlusonuçlarıdeneylerileldoğrulanmaktadır.Bununyanında, klasikikiboyutlu TDA tekniğine, üçüncüboyutolaraketiketleradapteedilmiĢvedeneyselsonuçlarraporlanmıĢtır.Bu iyileĢtirmelerdahadoğruvegeniĢletilebilirönerisistemlerioluĢturmayısağlamıĢtır.AnahtarKelimeler: ÖneriSistemleri, TekilDeğerAyrıĢımı, KolaboratifFiltreleme, ĠçerikBazlıFiltreleme, KiĢiselleĢtirme, KullanıcıModelleme

Every recommendation system follows a specific process while making recommendations. Systems use the users' profiles and the information about items or products as the inputs and produce recommendations. In other words, a recommendation system consists of background data, the information that the system has before the recommendation process begins, input data, the information that user must communicate to the system in order to generate a recommendation, and an algorithm that combines background and input data to arrive at its recommendations.

RECOMMENDER SYSTEMS

In order to increase the users' satisfaction towards online information search results, search engine developers and vendors try to predict user preference based on the user behavior. Recommendations are provided by the search engines or online vendors to the users. Recommendation systems are implemented in both commercial and nonprofit web sites to predict the user preferences. Accurate predictions may result in higher selling rates and increase the customer satisfaction. The main functions of recommendation systems are analyzing user data and extracting useful information for further predictions [6]. Variety of techniques has been proposed for performing recommendation, including content-based, collaborative, knowledge-based and other techniques. To improve performance, these methods have sometimes been combined in hybrid recommenders.

There are a lot of recommendation systems, accessible via internet, which attempt to recommend to users several products such as music, movies, books, etc. For instance, recommender systems are now an integral part of some e-commerce sites such as Amazon.com and CDNow [7]. In a general way, recommendation systems are systems which intend to acquire opinions or preferences about items from a community of users, and use those opinions to present other users with items that are interesting to them. From this general description we can see that recommendation systems need two basic things to work properly

In general, every recommendation system follows a specific process in order to create recommendations. If we see the process of recommendation as a black box, as shown in Figure 2.1, we can identify two sources of information needed as input for the process. These sources of information are the users' profiles and the information about items or products. Ideally the information stored in the profiles is related with the preferences of the users and should be given explicitly by the user itself. However, this information can also be extracted from other external sources such as web pages, buying behavior, etc.

REFERENCES

1. Ahmed N, Rao KR (1975) Orthogonal transforms for digital signal processing. Springer, New York Berlin HeidelbergCrossRefGoogle Scholar
2. Bourlard H, Kamp Y, Wellekens CJ (1985) Speaker dependent connected speech recognition via phonemic Markov models. Proc ICASSP, pp 1213–1216
3. Bunch JR, Nielsen CP (1978) Updating the singular value decomposition. Numer Math 31:111–129CrossRefGoogle Scholar
4. Cottrell GW, Munro PW, Zipser D (1988) Image compression by back propagation: a demonstration of extensional programming. In: Sharkey NE (ed) Advances in cognitive science, vol 2. Abbex, Norwood, (NJ) (in press)Google Scholar
5. Delsarte P, Kamp Y (1988) Low rank matrices with a given sign pattern Philips Research Laboratory, Brussels SIAM J: (to be published)
6. Elman JL, Zipser D (1987) Learning the hidden structure of speech. J AcoustSoc Am 83:1615–1626CrossRefGoogle Scholar
7. Golub GH (1968) Least squares, singular values and matrix approximations. ApplikaceMatematiky 13:44–51Google Scholar
8. Golub GH, Van Loan CF (1983) Matrix computations. North Oxford Academic, OxfordGoogle Scholar
9. Harrison TD (1987) A Connectionist framework for continuous speech recognition. Cambridge University Ph. D. dissertation
10. Lippmann RP (1987) An introduction to computing with neural nets. IEEE ASSP Magazine, pp 4–22
11. Minsky M, Papert S (1969) Perceptrons. MIT Press, CambridgeGoogle Scholar
12. Rumelhart DE, McClellarnd JL, and the PDP Research Group (1986) Parallel distributed processing. Exploration in the microstructure of cognition. vol 1–2. MIT Press, CambridgeGoogle Scholar
13. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning internal representations by error propagation. In: Rumelhart DE, McClellan JL (eds) Parallel distributed processing. Exploration in the microstructure of cognition, vol 1. Foundations. MIT Press, CambridgeGoogle Scholar
14. Stewart GW (1973) Introduction to matrix computations. Academic Press, New YorkGoogle Scholar