

# A Non-Parametric Data Envelopment Analysis Approach for Cloud Services Evaluation

Chunxiang Xu, Yupeng Ma<sup>(✉)</sup>, and Xiaobo Wang

XinJiang Technical Institute of Physics and Chemistry CAS, Beijing, China  
chunxiangxu@gmail.com, {ypma,wangxb}@ms.xjb.ac.cn

**Abstract.** Due to advantages of cloud computing, services are increasingly deployed in cloud. It is a challenge to choose a proper service. Besides QoS requirements, customers expect more efficient services which provide better performance but with minimum cost. In this paper, we propose a non-parametric method to evaluate relative efficiency of cloud services based on Data Envelopment Analysis. It can classify cloud services into different efficiency levels and tell how to improve less efficient services. We illustrate the method with a case study.

**Keywords:** Cloud service · Relative efficiency · QoS · Data envelopment analysis

## 1 Introduction

With the fast development of cloud computing, numerous services with similar functions have been presented to cloud customers at different prices and performances, therefore it is a challenge for customers to select the most “worthy” cloud services. Often, there may be trade-off between cost and performance which makes it difficult to evaluate services from different cloud providers in an objective way.

Lots of work have been done for comparing different cloud services. Some researchers specified factors or attributes important for evaluating cloud computing models [5, 10], such as security, availability, performance, cost, etc. Based on these attributes, cloud services are compared and ranked for selection.

Some of existing works [8, 11, 15, 16] compared services only based on performances while the cost was not considered. Some researchers considered both. Brebner [3] used statistical graphs to compare cost of different cloud scenarios. Li [9] proposed Cloudcmp to compare public cloud providers on different aspects separately. The work in [13] analyzed the application of different multicriteria decision analysis (MCDA) methods to cloud services selection, including AHP, TOPSIS and other methods taking multiple QoS attributes and cost into account. Saurabh [6] introduced a framework named SMICloud using AHP

---

Supported by the West Light Foundation of Chinese Academy of Sciences (Project No. XBBS201319).

method. It is parametric and requires customers to define the preference value for each factor first. However, sometimes we might only care about the minimum performance and the maximum cost, while the preference of QoS attributes is of little concern. In this case, the AHP method might not be appropriate, because different preference values will give different results.

DEA is a non-parametric method suitable for situation without user’s preference. In this paper, we introduce how to apply DEA to evaluate and compare a group of cloud services according to their prices and performances. According to features of DEA models and characteristic of cloud services, we combine results of C<sup>2</sup>R [4] and BC<sup>2</sup> [2], and define several efficiency levels for cloud services based on their relative efficiency scores and slacks. Thereby customers can select services of higher efficiency level. Moreover, for less efficient services, we show how they should be improved in order to be more efficient.

## 2 Classifying Cloud Services with DEA Models

### 2.1 DEA Models

Here we introduce how to construct DEA models for calculating relative efficiency scores and slacks. The problem can be defined as following: given  $n$  cloud services, each having  $s$  QoS attributes and  $m$  cost items, finding out which service’s QoS value could be improved with the cost remaining the same. The cloud services, QoS attributes and cost items are modeled as DMUs, output variables and input variables respectively.

The C<sup>2</sup>R and BC<sup>2</sup> models are used to calculate efficiency scores. The C<sup>2</sup>R model is designed with the assumption of constant returns to scale, meaning that if all inputs are doubled, the outputs are also expected to double. The BC<sup>2</sup> model takes into account that the performance at the most efficient point may not be attainable and thus assumes variable returns to scale.

The C<sup>2</sup>R model  $D_{C^2R}^I$  is given in Eq. 1, where  $x_{ij}$  and  $y_{rj}$  are constants representing the  $i$ th input and  $r$ th output for the  $j$ th service, and  $\theta$  and  $\lambda_j$  are variables of the LP problem. The BC<sup>2</sup> model  $D_{BC^2}^I$  is similar to  $D_{C^2R}^I$  with one more constraint  $\sum_{j=1}^n \lambda_j = 1$ . The optimal values of the two models  $h_{C^2R}^I$  and  $h_{BC^2}^I$  are called technical and scale efficient score (TSE score) and technical efficient score (TE score) respectively. It can be inferred that if TSE score is 1, then TE score is 1, but not vice versa.

$$D_{C^2R}^I \begin{cases} \min \theta = h_{C^2R}^I \\ \sum_{j=1}^n \lambda_j x_{ij} \leq \theta x_{i0} & i = 1, \dots, m \\ \sum_{j=1}^n \lambda_j y_{rj} \geq y_{r0} & r = 1, \dots, s \\ \lambda_j \geq 0 & j = 1, \dots, n \end{cases} \quad (1)$$

Because of the existence of weak efficient point which is on the efficient frontier but still possible to be improved, we apply the slack-based method [12] to calculate slacks. The model  $\bar{D}_{C^2R}^I$  expressed by Eq. 2 can be used to check if

a service is weak technical and scale efficient (W-TSE). Accordingly, a similar model  $\bar{D}_{BC^2}^I$  with one more constraint ( $\sum_{j=1}^n \lambda_j = 1$ ) can check weak technical efficient (W-TE). Here,  $\lambda_j, s_i^-$  and  $s_r^+$  are variables of the equations.  $s_i^{*-}$  and  $s_r^{*+}$  are values of  $s_i^-$  and  $s_r^+$ , the sum of which is the optimal value (called slack) of the model. If the slacks for the two models are greater than 0, the cloud service is called W-TSE and W-TE respectively.

$$\bar{D}_{C^2R}^I \begin{cases} \max \sum_{i=1}^m s_i^- + \sum_{r=1}^s s_r^+ = \sum_{i=1}^m s_i^{*-} + \sum_{r=1}^s s_r^{*+} & i = 1, \dots, m \\ \sum_{j=1}^n \lambda_j x_{ij} + s_i^- = h_{C^2R}^I x_{i0} & r = 1, \dots, s \\ \sum_{j=1}^n \lambda_j y_{rj} - s_r^+ = y_{r0} & j = 1, \dots, n \\ \lambda_j \geq 0 & i = 1, \dots, m \\ s_i^- \geq 0 & r = 1, \dots, s \\ s_r^+ \geq 0 & \end{cases} \quad (2)$$

**Table 1.** Service efficiency types and levels

Level	Type	$h_{C^2R}^I$	Slack_C <sup>2</sup> R	$h_{BC^2}^I$	Slack_BC <sup>2</sup>	Improvement
1	TSE	1	0	1	0	no
2	TE	<1	-	1	0	not sure
3	W-TSE	1	>0	1	-	yes
4	W-TE	<1	-	1	>0	yes
5	inefficient	<1	-	<1	-	yes

### 2.2 Efficiency Levels

Table 1 shows different efficiency types and their levels, where level 1 is the most efficient and level 5 is the least. Consider the cases of the second and the third row, if a cloud service is W-TSE, it can be improved by the distance of the slack. However, a TE service might not be able to be improved, because its improvement requires moving along the frontier of the C<sup>2</sup>R model and the efficient target might be an impossible situation for a cloud service. For example, the percent of SLA agreement is bounded to the range of [0,1] and it cannot be greater than 1 in order to be efficient. Therefore, if a cloud service is TE, we believe that its efficiency is better under current cloud services group. The order of efficiency level of 3, 4 and 5 is obvious. According to the above discussion, we can have the efficiency order defined in Table 1.

### 2.3 Method Implementation

Now we give detailed steps to evaluating cloud services' efficiency with DEA.

The first step is preparing data for the model. One thing is to check non-numeric value and missing value. Non-numeric value should be transformed to numeric value according to its definition. For example, if a service’s security protection level is specified as *poor*, *average* or *extensive*, these values can be transformed to integer values of 1, 2 and 3, with larger value meaning better performance. Missing value can be filled with the average value of the attribute or predict its value with a precise assumption [7, 14]. The other thing is to check QoS attributes. DEA generally minimizes “inputs” and maximizes “outputs”. In other words, smaller levels of the former and larger levels of the latter represent better efficiency. Therefore, for attribute having smaller value for better performance, such as response time, we use the reciprocal of the variable instead.

The second step is to determine efficiency level of a cloud service based on its efficiency scores and slacks. First we calculate the TSE score with the model  $D_{C^2R}^I$ . If it is equal to 1, the model  $\bar{D}_{C^2R}^I$  is used to obtain the slack. If the slack is 0, then the cloud service is of level 1, otherwise it is of level 3. If the TSE score is smaller than 1, we continue to solve the model  $D_{BC^2}^I$ . If the TE score is smaller than 1, the service is inefficient and of level 5. Otherwise, the model  $\bar{D}_{BC^2}^I$  is applied to get the slack of  $BC^2$  model. The service is of level 2 if the slack is 0, otherwise it is of level 4.

The last step is to obtain the improving targets of less efficient cloud services by calculating their projections on frontier of either  $D_{C^2R}^I$  model or  $D_{BC^2}^I$  model. If a cloud service’s efficiency is of level 2 or 3, its projection can be obtained according to Eq. 3, where  $s_i^{*-}$  and  $s_r^{*+}$  are from the optimal value of model  $\bar{D}_{C^2R}^I$ . If the efficiency level of the cloud service is 4 or 5, its projection can be obtained according to Eq. 4, where  $s_i^{*-}$  and  $s_r^{*+}$  are from the optimal value of model  $\bar{D}_{BC^2}^I$ .

$$(P - j_0) \begin{cases} \hat{x}_{i0} = h_{C^2R}^I x_{i0} - s_i^{*-}, & \text{for } i = 1, \dots, m \\ \hat{y}_{r0} = h_{C^2R}^I y_{r0} + s_r^{*+}, & \text{for } r = 1, \dots, s \end{cases} \quad (3)$$

$$(P - j_0) \begin{cases} \hat{x}_{i0} = h_{BC^2}^I x_{i0} - s_i^{*-}, & \text{for } i = 1, \dots, m \\ \hat{y}_{r0} = h_{BC^2}^I y_{r0} + s_r^{*+}, & \text{for } r = 1, \dots, s \end{cases} \quad (4)$$

### 3 Case Study

We illustrate our method by evaluating relative efficiency of a group of IaaS services from an online data set [1]. Due to limited pages, we select 15 services from three cloud providers. The input data and analysis results are listed in Table 2, where p1s1 corresponds to the first service from the first provider, and (0.12,1,2,3.75,410) represents values of price/hour, number of virtual cores, compute units, memory(GB) and disk(GB).

It can be seen that services from the first and second providers are more efficient. Projections for services of level 1 are the same as their original values, while others are targets for improving services’ performance in order to become efficient. For example, the projection of service p2s2 is (0.12,4,4,4,80), whose

**Table 2.** Case study result

Service	$h_{C^2R}^I$	$h_{BC^2}^I$	Slack_C <sup>2</sup> R	Slack_BC <sup>2</sup>	Input data Projection	Level
p1s1	0.9982	1.0000	-	0.1304	(0.12,1,2,3.75,410) (0.1198,1.0797,2.0507,3.75,410)	4
p1s2	1.0000	1.0000	0.0000	-	(0.24,2,4,7.5,840)	1
p1s3	1.0000	1.0000	0.0000	-	(0.48,4,8,15,1680)	1
p2s1	1.0000	1.0000	0.0000	-	(0.03,1,1,1,20)	1
p2s2	1.0000	1.0000	2.0000	-	(0.12,2,4,4,80) (0.12,4,4,4,80)	3
p2s3	1.0000	1.0000	4.0000	-	(0.24,4,8,8,160) (0.24,8,8,8,160)	3
p3s1	0.8476	0.8476	-	2.9513	(0.12,1,1,3.35,123) (0.1017,1.8854,3.0660,3.35,123)	5
p3s2	0.9908	0.9908	-	3.0248	(0.24,2,2,7.5,738) (0.2378,2.5075,4.5174,7.5,738)	5
p3s3	0.9904	1.0000	-	6.7783	(0.48,4,4,15,1467) (0.4754,5.6978,9.0804,15,1467)	4
p4s1	0.7500	0.7500	-	0.0000	(0.04,1,1,1,20) (0.03,1,1,1,20)	5
p4s2	0.8063	0.9667	-	0.1875	(0.08,2,2,2,60) (0.0773,2,2.05,2.1375,60)	5
p4s3	0.7500	1.0000	-	0.0000	(0.16,4,4,4,80) (0.12,4,4,4,80)	2
p5s1	0.3113	0.3115	-	0.0481	(0.1,1,1,1,25) (0.0312,1,1.0129,1.0353,25)	5
p5s2	0.3675	0.4267	-	0.5625	(0.2,2,2,2,100) (0.0853,2,2.15,2.4125,100)	5
p5s3	0.4150	0.5467	-	0.1875	(0.3,4,4,4,100) (0.164,4,4.05,4.1375,100)	5

input data is (0.12,2,4,4,80), thus its number of virtual cores should be changed from 2 to 4, and the distance between the two value (4−2=2) is slack for virtual cores. From all the slack values, we can see that services from the third provider need more improvement.

### 4 Conclusion and Future Work

In this work, non-parametric method based on DEA model is applied to evaluate the relative efficiency of cloud services, which is the ratio of performance and price. The method can evaluate and classify cloud services by analyzing their efficiency, and further show how to improve inefficient services.

Our work can classify a group of cloud services according to their efficiency levels, but services of the same level cannot be distinguished yet. Next, we will study how to distinguish services of the same efficiency level so as to offer more support for user’s decision.

### References

1. IaaS cost snapshot, 29 May 2014. <https://docs.google.com/spreadsheets/fm?id=t9fsKqXYKmiQ6LFvmVkHTCQ.04776183464924915040.7277648387216344348&fncmd=420>
2. Banker, R.D., Charnes, A., Cooper, W.W.: Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Manag. Sci.* **30**(9), 1078–1092 (1984)

3. Brebner, P., Liu, A.: Performance and cost assessment of cloud services. In: Maximilien, E.M., Rossi, G., Yuan, S.-T., Ludwig, H., Fantinato, M. (eds.) ICSOC 2010. LNCS, vol. 6568, pp. 39–50. Springer, Heidelberg (2011)
4. Charnes, A., Cooper, W.W., Rhodes, E.: Measuring the efficiency of decision making units. *Eur. J. Oper. Res.* **2**(6), 429–444 (1978)
5. Consortium, C.S.M.I.: Service measurement index framework version 2.0 (2014). [http://csmic.org/wp-content/uploads/2014/01/SMI\\_Overview\\_140113.pdf](http://csmic.org/wp-content/uploads/2014/01/SMI_Overview_140113.pdf)
6. Garg, S.K., Versteeg, S., Buyya, R.: A framework for ranking of cloud computing services. *Future Gener. Comput. Syst.* **29**(4), 1012–1023 (2013)
7. Goldman, A., Ngoko, Y.: On graph reduction for qos prediction of very large web service compositions. In: 2012 IEEE Ninth International Conference on Services Computing (SCC), pp. 258–265 (2012)
8. Huang, K., Yao, J., Fan, Y., Tan, W., Nepal, S., Ni, Y., Chen, S.: Mirror, mirror, on the web, which is the most reputable service of them all? In: Basu, S., Pautasso, C., Zhang, L., Fu, X. (eds.) ICSOC 2013. LNCS, vol. 8274, pp. 343–357. Springer, Heidelberg (2013)
9. Li, A., Yang, X., Kandula, S., Zhang, M.: Cloudcmp: comparing public cloud providers. In: Proceedings of the 10th Annual Conference on Internet Measurement, pp. 1–14 (2010)
10. Parhizkar, B., Abdulhussein, A.A., Joshi, J.H., Twinamatsiko, A.M.: A common factors analysis on cloud computing models. *Int. J. Comput. Sci. Issues* **10**(2), 523–529 (2013)
11. Park, J., Jeong, H.Y.: The QoS-based MCDM system for SaaS ERP applications with social network. *J. Supercomput.* **66**(2), 614–632 (2012)
12. Tone, K.: A slacks-based measure of efficiency in data envelopment analysis. *Eur. J. Oper. Res.* **130**(3), 498–509 (2001)
13. Whaiduzzaman, M., Gani, A., Anuar, N.B., Shiraz, M., Haque, M.N., Haque, I.T.: Cloud service selection using multicriteria decision analysis. *Sci. World J.* **2014**, 10 (2014)
14. Xie, Q., Wu, K., Xu, J., He, P., Chen, M.: Personalized context-aware QoS prediction for web services based on collaborative filtering. In: Cao, L., Zhong, J., Feng, Y. (eds.) ADMA 2010, Part II. LNCS, vol. 6441, pp. 368–375. Springer, Heidelberg (2010)
15. Zheng, Z., Wu, X., Zhang, Y.: QoS ranking prediction for cloud services. *IEEE Trans. Parallel Distrib. Syst.* **24**(6), 1213–1222 (2013)
16. Zheng, Z., Zhang, Y., lyu, M.R.: Cloudrank: a qos-driven component ranking framework for cloud computing (2010)