Dynamic Pricing Strategy for Cloud Computing with Data Mining Method

Xing Wu[1,2,3], Ji Hou[1], Shaojian Zhuo[1], and Wu Zhang[1]

[1] School of Computer Engineering and Science,
Shanghai University, Shanghai, China
[2] State Key Laboratory of Software Engineering,
Wuhan University Wuhan, China
[3] State Key Laboratory for Novel Software Technology,
Nanjing University Nanjing, China
xingwu@shu.edu.cn

Abstract. Cloud computing is the delivery of computing as a service rather than a product, whereby shared resources, software, and information are provided to computers and other devices as a metered service over a network (typically the Internet). To maximize the revenue of cloud service providers, a dynamic pricing model is proposed, which consists of two data mining methods. The first data mining method is the k-means algorithm with which historical data are classified into groups. The second one is Bayes decision that can forecast the trend of user-preferred cloud service packages. In proposed pricing model, BP-neutral network is applied to forecast the price which can maximize the revenue. Compared with the static pricing model and the models without k-means algorithm, the proposed model can meet customers' demand better and outperform them in revenue maximization.

Keywords: maximum revenue, various resources, dynamic pricing, Cloud Computing.

1 Introduction

Nowadays, more and more users are concentrating on Clouds Computing systems. To cater to these demands, some corporations have implemented cloud computing to go along with the new technology, such as Google and Amazon. Google engaged in the development of cloud computing technology and proposed Google File System (GFS), MapReduce and Bigtable[1]. While Amazon EC2 (Elastic Compute Cloud) has successfully gain profit from its cloud services. To maximize the revenue [2] of cloud providers, a proper pricing strategy is indispensable.
Other corporations have developed their own cloud platforms too. For instance,

As the number of cloud users varies from time to time and the demands of users fluctuate, a dynamic pricing model to manage the revenue is more effective than the static pricing model. Thus a dynamic pricing model implementing the neutral

Y. Zhang, K. Li, and Z. Xiao (Eds.): HPC 2012, CCIS 207, pp. 40–54, 2013.

networks to make a forecast according to requests made by users is proposed to maximize the revenue of cloud providers. In the proposed model we also forecast the user-preferred pattern [3] of cloud service packages using the Bayes decision. Furthermore, some data mining methods are utilized to classify the customers based on their behaviors.

2 Related Works

2.1 The Definition of Cloud Computing

Cloud Computing has a strong relationship with Grid Computing. Although there are so many relations between these two kinds of styles, it is the differences that exist. Although there are lots of definitions for cloud computing, we still don't have a universally agreed definition. A popular definition for cloud was proposed by Ian Foster as follows: A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet [4]. As can be seen in this definition, the most important developed trends in cloud that the Grid Computing doesn't have are virtualized, dynamically-scalable platforms and services which are delivered on demand to external customers over the Internet. Since the resources have been virtualized as commodities, the transactions in cloud computing should follow the law of economics. Compared with Grid computing, the services on Cloud Computing can even be virtualized as commodities which makes the cloud computing much easier to be charged.

Amazon EC2 charges for not only the cloud infrastructures but also cloud services. An elastic model is proposed in the EC2 which means cloud can be seen as resources without a boundary [5]. To extend this concept, cloud computing should provide services without a boundary and thus should be charged dynamically.

2.2 The Revenue Management of Cloud Computing

As the pricing becomes an indispensable part of the cloud computing, it is a problem how to make the profits maximum. Some scholars analyzed the cost-benefit of Cloud Computing [6] and believed cloud computing has advantages in following aspects: the reliability, the pricing and the response time concerned with the service quality. With the cloud computing becoming widely used, pricing could play the vital role to make the revenue maximum which is related to the revenue management of any company. The term revenue management is most commonly used for the theory and practice of maximizing expected revenues by opening and closing different fare classes or dynamically adjusting prices for products. Putting this conception into clouding pricing is meaningful. In the revenue management of cloud computing, a dynamic pricing strategy is required to maximize the revenue.

2.3 The Dynamic Pricing of Cloud Computing

There has been some scientific research concerned with dynamic pricing models for cloud computing. Arun Anandasivam and Marc Premm proposed both a static pricing model and a dynamic pricing model which utilizes a heuristic algorithm to forecast the pricing [7]. In this research, the static model can't be adjusted to changing situations, whereas the proposed dynamic pricing model adapts to the real cloud market. However the proposed dynamic pricing model only contained a math model without an implementation and not available for multiple kinds of cloud resources. Furthermore the model didn't make revenue maximum. So we make an improvement for multiple kinds of cloud resources.

There is a big difference between one kind of resource and multiple kinds of cloud resources for dynamic pricing. For multiple kinds of cloud resources, the relationship among these resources should be considered. To simply the situation of multiple cloud resources, "combo" and "package" are defined. Combo is the combination selected by customers, while package is the combination predetermined by company previously.

Furthermore, a data mining method is used to classify the historical data which is used for forecasting. Different people have different demands. With this method the customers can be put into different classifications, so that a more specific forecast can be made using the sorted data.

When a forecast for price adjustment is made, there should be a time scale. In [7], time interval is the second-scale period which is too precise to realize it. Thus time scale in hours is used in our pricing model. Before a forecast for the combos is made, a data mining method is proposed to classify the various resources into several classifications as different customers have different requests. Then the most similar historical data can be utilized to forecast future usage with back-propagation neutral network approach [8]. As the fluctuation of price has no regular patterns, the back-propagation neutral network approach is appropriate for this situation [9]. Back-propagation neutral network is effective when there is no obvious rule [10]. After the forecast of back-propagation neutral network, a matrix of prices and amount of requests which are accepted by customers will be gotten. Then an equation of the maximum revenue can be acquired. After we seek partial derivative for each resource price components, extreme points can be sought and then find the max point which is to say to find the maximum revenue. Meanwhile the dynamic pricing is determined.

As the conception that cloud is infinite, we don't need to consider remained resources or resources [11] that will be set free to adjust the price as other traditional industries.

When we forecast the trend of packages [12], the time scale will be much longer, such as time scale of months which is more reasonable. In this part, a way to forecast the trend of using situation of packages is proposed and it is especially applied for the packages.

3 Pricing Determination in Various Cloud Resources

3.1 Scenario

When some cloud resources are priced, first of all, what kind of services is used should be decided, for example, a package or a combination, and then the using time, which is to say, how long the resources need to be used, which decides the time scale in our dynamic pricing model. In *Dynamic Pricing* how various resources that are chosen by customers themselves are priced is going to be talked about so as to get a biggest profit. In the next *Forecast Model* how to forecast the trend of using situation of different packages will be talked about, in which the contents of packages have been decided by the operating agencies.

In *Dynamic Pricing*, the time scale is a comparatively short time. It can be defined as hours, while in *Forecast Model* the time scale will be much longer, and it can be defined as several days even several months. Because the package is more static than the combination, once the price and contents of the package is defined, it will not change until next season or next time when the company decides to make a change, while the combination is always changing as customers have different requests.

3.2 Basic Model and Dynamic Pricing

There is a certain amount of packages as $\mathbf{X}_{i,\, i=1,2,3\ldots\ldots}$. Each $\mathbf{X}_i$ includes different resource types of uncertain amount. Different resource types in a package can be defined as a vector $\mathbf{X}_i = (x_1, x_2, x_3 \ldots\ldots x_k)$ where x_k represents a kind of resource such as CPU or MEM and so on. $\mathbf{X}_i$ can also represent resource combinations which are made by customers themselves in *Dynamic Pricing*. User requests (what have been accepted, the below is the same) are always made over times. T represents the spot when a user request is made.

When pricing, not only the present price, but also the change of request trend is necessary. If there are more requests in the future, it can indeed influence pricing. The customers' behaviors can be thought as a serious of changing requests. Thus a metric is defined to represent the trend of changing combination amounts from then to future. As the follows:

Table 1. Trend of Changing Combination Amounts

...	t_i	t_{i+1}	t_{i+2}	t_{i+3}	...	$T-2t$	$T-t$	T	$T+t$	$T+2t$	...	t_j	t_{j+1}	...
...	X_1	X_1	X_1	X_1	...	X_1	X_1	X_1	X_1	X_1	X_1	X_1	X_1	X_1
...	X_2	X_2	X_2	X_2	...	X_2	X_2	X_2	X_2	X_2	X_2	X_2	X_2	X_2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	..
...	Xn_i	Xn_{i+1}	Xn_{i+2}	Xn_{i+3}	...	X_{n-2t}	X_{n-t}	X_n	X_{n+t}	X_{n+2t}	...	Xn_j	Xn_{j+1}	...

In Table 1, T represents the present time, t_i represents a past time while t_j represents a future time. The period of t is not defined which can be chosen by the real situation. For example, t can be 1 hour or half an hour, and so on. And when pricing, both historical amount of accepted requests and relevant price should be considered. With the historical data mentioned before, the accepted requests and the relevant price in the future can be forecasted by BP Neural Network algorithm.

Through adjusting the price of different kinds of resources in the future, a relevant request amount in the future can be forecasted by BP Neural Network. Then the revenue could be maximized by the method which will be mentioned later.

If only one resource is put into consideration in dynamic pricing model, it can be represented as a single symbol, but several resource types for a dynamic pricing model are different. It has to be concerned with a matched problem. And different resource types can form a large quantity of combinations

There are i historical spots considered to price a combination. As cloud resources are by some means combined, the used historical data should be similar with the combination which is priced. Thus the historical data could be classified into different classifications.

In order to be quickly calculated and self-match, which is more appropriate for the dynamic pricing model and more convenient, the algorithm K-means which is one of data mining methods is proposed as following pseudo code:

Algorithm1. K-means

Required: past data and request now dealt as

$\mathbf{X} = \{X_1, X_2 \ldots\ldots X_n, X_{request}\}$

(1)Define c as classification number.

(2)Define the permissible error as ε and define t=1;

(3)Initializing cluster centers $w_i(t), i = 1,2,\ldots,c.$ where c centers are chosen from X random.

(4)Sum of squared error criterion function is as the target function:

$$J_e = d_{ij}\sum_{i=1}^{c}\sum_{j=1}^{n}\| x_j - w_i \|^2 \quad d_{ij} = \begin{cases} 1, x_j \in w_i \\ 0, x_j \notin w_i \end{cases}$$

(5)Modified clustering center:

$$w_i(t+1) = \frac{\sum_{j=1}^{n} d_{ij}x_j}{\sum_{j=1}^{n} d_{ij}}, i = 1, 2, \ldots, c.$$

(6)Calculate the error:

$$E = \sum_{i=1}^{c}\| w_i(t+1) - w_i(t) \|$$

(7)***If*** E< ε ***then***

For i=1 to c

Use the Je to judge to which group the $X_{request}$ belongs

If je is minimum ***then***

Answer =i;

Endif

Endfor

else t=t+1 goto (4)

Endif

"Answer" is the classification the present request belongs to. Thus the historical data in classification "Answer" is considered to be used in the Back Propagation Neural Network. The Back Propagation Neural Network can be defined as function network(x) which will be used in maximizing revenue.

Before using network(x), the data should be dealt with, so as to match with network(x). $\mathbf{P_i}(p_1 \quad p_2 \quad \ldots \quad p_k)$ represents the price of each $\mathbf{X_i}$ of chosen classification and then each price of $\mathbf{X_i}$ should be calculated by unit as $\mathbf{P_i}$. After that there is a new $\mathbf{P_i}(p_1 \quad p_2 \quad \ldots \quad p_k)$ in which p_k is all by unit, x_{ik} is amount of each resource in $\mathbf{X_i}$.

$$\mathbf{P_i}(p_1 \quad p_2 \quad \ldots \quad p_k) = \mathbf{P}_j\begin{pmatrix} \frac{p_1}{x_{i1}} & \frac{p_2}{x_{i2}} & \ldots & \frac{p_k}{x_{ik}} \end{pmatrix} \tag{1}$$

And then a Price demand prediction analysis matrix can be defined as following, in which each row is the new $\mathbf{P_i}$ and the last col is the sum amount of all the accepted requests as r_i

$$\begin{pmatrix} p_{11} & p_{12} & \ldots & p_{1k} & r_1 \\ p_{21} & p_{22} & \ldots & p_{2k} & r_2 \\ p_{31} & p_{32} & \ldots & p_{3k} & r_3 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ p_{n1} & p_{n2} & \ldots & p_{nk} & r_n \end{pmatrix} \tag{2}$$

As to pricing, a dealt algorithm can be defined as mentioned in the previous section. What is predicted is the trend of users' behaviors, thus some previous time nodes' data is used to forecast the future spot's users' behavior, which is to say, if the users will accept the price. The process before is to make the price unit, but the amount of visitors is also different, thus making them in a same standard is necessary. For example, at 1st hour there are 1000 visitors and 800 people accept the price which is to say there are 800 accepted requests and price p1, at 2nd hour there are 100 visitors and 90 people accept the price, which is to say there are 90 accepted requests and price p2 that is less than p1. Even though p2 is less, price p1 attract more accepted requests, because the visit quantity of 1st hour is more than 2nd hour. When a forecast of customers' behavior is made, a unified standard is necessary, because the customers will not care how many people visit the website at a spot, what they care is only the price. And considering the resource amount is different in each request, the sum of each resource amount should be used not the request amount. In order to solve those problems, which can influence the forecast of customers' behaviors, the following algorithm is proposed:

At a certain time T, there are several requests for different combinations. S is defined as a vector in which each component represents the sum of each resource amount in a data center as $\mathbf{S}=(s_1, s_2, s_3, s_4 \ldots\ldots s_n)$. And the vector $\mathbf{X_i}$ is the request of that spot. Each component of $\mathbf{X_i}$ represents each resource amount. Then all resources of requests should be summed by adding the $\mathbf{X_i}$ up as

$$\mathbf{Sx}=\sum_n \mathbf{X}_i \text{ , } \mathbf{X_i} = (x_1, x_2, x_3 \ldots\ldots x_k) \tag{3}$$

If $\mathbf{Sx}$ is not matched to S, a special augmented matrix of $\mathbf{Sx}$ can be defined as $\widetilde{\mathbf{Sx}}$ in which a zero is added to where there is no request of this kind of resource to make sure that number of Sx's cols equal S's. Thus $\mathbf{Sx/S}$ will be a part of the equation to price in a same standard. As it is known two vectors can't make a division, thus some little adjustments can be made and do not change the equation's meaning. A diagonal matrix for S like the style:

$$\text{diag}(s_1 \quad s_2 \quad \dots \quad s_3) = \begin{pmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_n \end{pmatrix} \tag{4}$$

In the equation an inverse matrix $\mathbf{S}^{-1}$ is needed for diag($s_1 \quad s_2 \quad \dots \quad s_3$) to make calculation. Thus $\mathbf{Sx/S}$ can be thought as another form $\mathbf{Sx}\times\mathbf{S}^{-1}$. And percentage that amount of resources of accepted requests in sum of resources at a spot multiplies the percentage that accepted requests $R(t)$ in amount of visitors as $V(t)$. So a function to calculate at each spot the relationship between price and the resource amounts can be made.

$$\psi(t) = \frac{R(t)}{V(t)} \times \frac{\mathbf{Sx}(t)}{\mathbf{S}} = \frac{V(t)}{R(t)} \times (\mathbf{Sx}(t) \times \mathbf{S}^{-1}) \tag{5}$$

At last, a forecast should be made by Back Propagation Neural Network algorithm with training data of historical data and simulation as the form (10).General number of hidden layer neuron to determine the empirical formula is:

$$i = \sqrt{m+n} + a \tag{6}$$

Where I is the number of hidden layer neurons, n is the number of input layer neuron. The learning regular of neutral network is to confirm a W to make the error minimum.

$$F(W) = (D-T)^T (D-T) \tag{7}$$

And $\mathrm{D} = (d_1 \quad d_2 \quad \dots \quad d_n)$ is the idea output, if D is the output of the last layout of neutral network then n will be the number of output. The W can be made by several tries as

$$W_k(t+1) = W_k(t) + \delta W_k(t) \tag{8}$$

Where k means the order of layouts of neutral network and δ means the adjust value. The neutral network will correct the error of the ideal output and the normal output from the back to the front until the error is the minimum. When the neutral network makes the error minimum, it will be along the decrease which is most fast as

$$\delta W_k(t) = \left(\frac{\partial F(W)}{\partial w_{ij}}(t) \right)_{n_{k-1} \times n_k} \tag{9}$$

After the structure of the neural network is decided, the input vector and the output vector are going to be designed. There is input vector as {price of various resources}, where accepted request is from $\psi(t)$. And output vector as {a standard calculated by (5)}

$$\mathbf{P} = \begin{pmatrix} p_1 & p_1 & \dots & p_k \\ p_2 & p_2 & \dots & p_k \\ \dots & \dots & \dots & \dots \\ p_{n1} & p_{n2} & \dots & p_{nk} \end{pmatrix} \quad \mathbf{T} = \begin{pmatrix} \psi(t_{n1}) \\ \psi(t_{n2}) \\ \dots \\ \psi(t_{ni}) \end{pmatrix} \tag{10}$$

And in neutral network the Excitation function is defined as

$$f(x) = \frac{1}{1+\ell^{-x}} \tag{11}$$

Because f(x)<1 some adjustments of input P and output T are made to make them less than 1. They can be divided by a certain number which depends on real situation. Then the network can be trained as network(x) whose input is price and output is the percentage calculated in (5).

Algorithm 2. BP Neural Network algorithm In Matlab

```
%Training net (just a template)
clear all
function net=network(P,T)
S1=11 according to (4);
S2=1;
net = newff(minmax(P),[S1 S2],{'tansig','logsig'},'traingdx','learngdm');
net.performFcn='sse'
net.trainParam.show=1;
net.trainParam.mc=0.95;
net.trainParam.epochs=50;
net.trainParam.goal=0.001;
[net,tr]=train(net,P,T);
```

After training, the simulation can be made. Also the input should be made less than 1 by dividing them with a number and let the output multiply the same number. The output is the symbol of accepted percentage like (5) forecasted by neural network. If a higher price is not expected because there will be a risk that the higher price may make a fewer revenue, a network(0) is proposed or even a lower price to attract more accepted requests. As to how to set the price which is put into network(x), a price range maybe between ra=-10%~10% which is just a suggestion can be taken into the neural network as network (x×(1+ra)). Following algorithm will finally decide the price:

Algorithm 3. Making revenue maximum with discrete method

n←number of resources
Mr←0
Pr←0
For i=1 to n
For x=-10% to 10%
$P_i = P_i \times (1+x)$
$r_i = network(x)$
If $P_i \times r_i$ >Mr ***then***
Mr= $P_i \times r_i$
Pr= P_i
Endif
Endfor
Endfor

Then the result Mr represents the maximum revenue in some degree, and the price Pr is the price. When a forecast is made using Back Propagation Neural Network algorithm with proper training, as it can be seen in Fig.1, in only several times the goal can be reached. The one of the simulation in neutral network is as following figure:

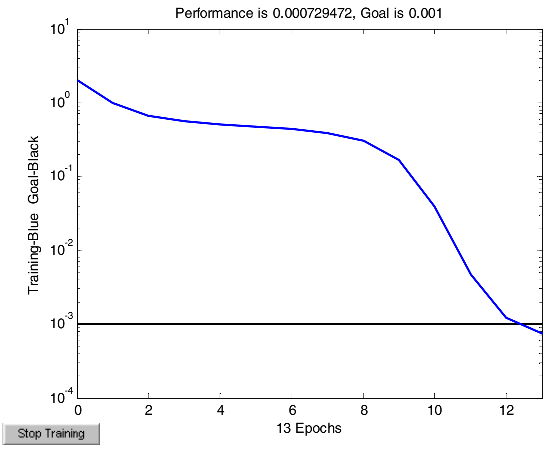

Fig. 1. Performance of BP Neutral Network

But there is still a problem about the way in which revenue is maximized. Price adjustment is discrete. If the maximum revenue being more accurate is expected, a continuous method is proposed. Firstly, the Back Propagation Neural Network algorithm to forecast the relationship between price and the accepted requests as the way mentioned before is also necessary. The equation of max revenue can be defined as follows:

$$y = p_1 \times network(p_1) + p_2 \times network(p_2) + \ldots\ldots p_i \times network(p_i) \tag{12}$$

Where network(pi) means the forecast result of neural network and pi is the price of this kind of the resource, so y is the revenue. Secondly, a linear regression to fit a function of price and the requests corresponding to the price is proposed to find the relationship between pi and network(pi). And a proper nonlinear regression model is better than the linear regression. Then (12) can be changed as the follows:

$$y = f(p_1) + f(p_2) + \ldots\ldots f(p_i) \tag{13}$$

Thirdly, partial derivative of y respect to each pi is calculated separately.

$$\frac{\partial y}{\partial p_j} = \frac{\partial f(p_1)}{\partial p_j} + \frac{\partial f(p_2)}{\partial p_j} + \ldots\ldots \frac{\partial f(p_i)}{\partial p_j} \quad j=1、2、3\ldots\ldots I \tag{14}$$

When $\frac{\partial y}{\partial p_j} = 0$ the corresponding price of each kind resource is gotten, then the maximum revenue is also achieved.

3.3 Forecast Model

As it is known, cloud resources can not only be as a combination, but also be sold as a package. A package is more static than a combination since the combos are always changing. Thus when a forecast model is made for packages, a longer time scale than the combo is necessary. The time scale in this part can be a week, a month, even half a year.

And when there is a package which is not priced or not be introduced, whether it is popular in the daytime or night is still unknown. Thus a Bayes to predict its using trend is needed so that a good price can be made. Even though it may have already been introduced, the use trend of it is not known exactly, because it can't reflect people's real desire that which kind of package they want, since there is not a really desired package for the customers, they only can buy a kind of package which is most closed to their desired package. Using Bayes model to forecast can help us know what the customers want much more specific, which is to say, at which time what users want exactly can be known.

And in this forecast model there is only a forecast of the trend of packages' using situation not pricing. This part is not like *Dynamic Pricing*. Langer time scale will take a pricing problem in economy. For example in *Dynamic Pricing* time scale is 1 hour, in this one hour the prime cost of MEM will not change so much but in a longer time scale as 1 month it's hard to say. Thus a forecast is just made for trend not pricing dynamically. In this part, a Bayesian Decision Theory to forecast is proposed when the package will be used more and its possibility.

For example, there are past 1 month data which are the data of combination not the data of packages because data of packages is of no value for customers' really desired packages and when there is a wish to make a new kind of package, the data of combo will be used. The data then are divided into three time periods as Morning (6:00-14:00), Afternoon (14:00-22:00) and Night (22:00-next day 6:00).

After that a Distribution model should be determined such as Normal distribution or T distribution or Γ distribution. There a Normal distribution is proposed as it's easy to understand.

Now there are several packages as $\mathbf{X_i} = (x_1, x_2, x_3 \ldots\ldots x_k)$ where x represent each different resource. 1 month data is divided by three time period as follow:

Table 2. Data of Combinations in a Month

	Day1	Day2	Day3	……	Day30	Day31
Morning	M1	M2	M3	……	M30	M31
Afternoon	A1	A2	A3	……	A30	A31
Night	N1	N2	N3	……	N30	N31

where A_1 means all the accepted combination in Day 1 Afternoon. N and M are the same. And $\mathbf{A_1} = (\mathbf{X_1} \ \mathbf{X_2} \ \ldots \ \mathbf{X_n})$ where each $\mathbf{X_i} = (x_1, x_2, x_3 \ldots\ldots x_k)$ in A means a combination, n is the amount of combinations. First the three time period's average and the standard deviation are calculated.

Morning:

$$ans_i = \frac{\sum_j (x_i \in \mathbf{X}_j)}{n} \qquad \mathbf{ans} = (ans_1 \quad ans_2 \quad \ldots \quad ans_i) \tag{15}$$

$$stdM = \frac{\sum_{1}^{31} \mathbf{ans}_i}{31} \tag{16}$$

$$\mathrm{s}M = \sqrt{\frac{[(\mathbf{X}_1 - \mathbf{std}M)^2 + (\mathbf{X}_2 - \mathbf{std}M)^2 + \ldots\ldots(\mathbf{X}_n - \mathbf{std}M)^2]}{n}} \tag{17}$$

The Afternoon and night are the same. Then the three groups of average and standard deviation are taken into the equation of Multidimensional Normal distribution

$$c \bullet \exp\left(-\frac{1}{2R}\sum_{i,j=1}^{m} R_{ij}\frac{x_i}{\sigma_i}\frac{x_j}{\sigma_j}\right), c = (2\pi)^{-\frac{m}{2}}(\sigma_1 \ldots \sigma_m)^{-1} R^{-\frac{1}{2}} \tag{18}$$

And R is the determinant of Correlation matrix (p_{ij}), R_{ij} is its Algebra formula. Then the three Normal distributions will be simultaneous to figure out two Demarcation point or interface. The Bayes formula is used:

$$\mathbf{P}(\mathbf{B} \mid \mathbf{A}) = \frac{\mathbf{P}(\mathbf{B}) \times \mathbf{P}(\mathbf{A} \mid \mathbf{B})}{\mathbf{P}(\mathbf{A})} \tag{19}$$

In details there is a kind of package X, then

$$\mathbf{P}(w_i \mid \mathbf{X}) = \frac{\mathbf{P}(w_i) \times \mathbf{P}(\mathbf{X} \mid w_i)}{\mathbf{P}(\mathbf{X})}, \quad w_i \in \{Morning \quad afternoon \quad night\} \tag{20}$$

Prior probabilities can be the same or based on the real situation. Thus that when the package is more popular can be known, that is the forecasted trend, and its relative probability.

When there are more people want to use this kind of package, maybe the package can be a higher price. As how to make the revenue maximum, you can have the trend which is forecasted by the Bayesian model, combined with a specific economic model.

Through Multidimensional normal distribution there is a basic figure.

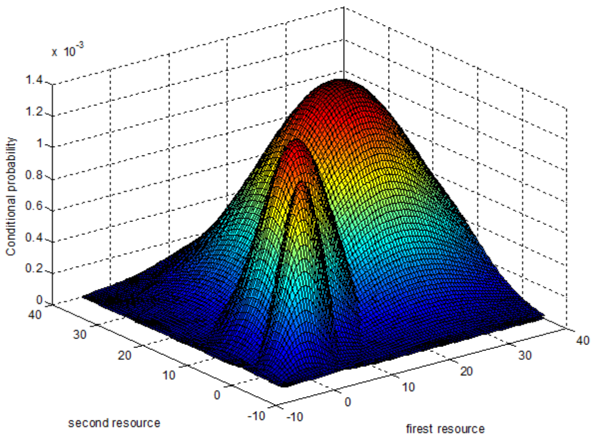

Fig. 2. The 3D view of Bayesian Decision

Then there is a vertical view to look at a Demarcation interface.

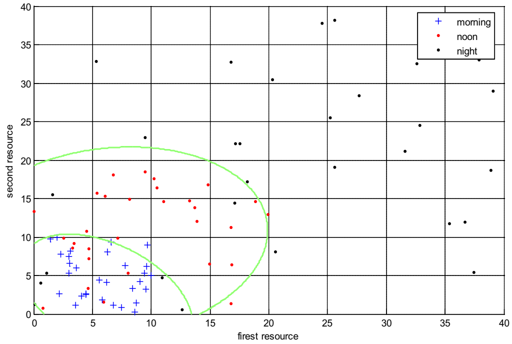

Fig. 3. Vertical view the 3D view of Bayesian Decision

The package which is forecasted can be made into the vertical view which is the Demarcation interface to look up its possibility in each time period. As can be seen from the vertical view of Bayes Decisions, in the morning people more like the package with first resource 5 and second resource 5 around. And in the afternoon people more like the package with first resource 15 and second resource 15 around while in the evening people's requests are more dispersed between 25 to 40 of both resources although there are several requests of night is between 5 to 20(the black dot), it is less possibility. So the price according to the forecast trend can be made.

4 Numerical Studies

In this section, numerical studies are conducted to evaluate the results between the dynamic pricing model and the static pricing model. The request is set by a Normal distribution. Dynamic pricing model is realized in Matlab and make a comparison.

4.1 Comparison between Static Model and Dynamic Model

In this section, there is a comparison about the revenue between our dynamic pricing model and the static pricing model with two various resources as A and B. In Fig.1 it is the comparison of prices between the dynamic pricing model and the static pricing model. As it can be seen, the static pricing is always 10 which is the black line while the dynamic price is between 11 and 9. With the dynamic price changing there are different requests accepted as in Fig.2. In Fig.2 it can be seen that the requests of static pricing model which is the blue line are corresponding to the Normal distribution as mentioned before while the requests of our model is more or less. But no matter the requests of our model is more than the static pricing model or less, the revenue is more in our model as in Fig.3 and Fig.4. In Fig.3 the time scale is hours, the revenue is summed of each hour in continues 24 hours. In Fig.4, the revenue of each day is summed in continues 31 days.

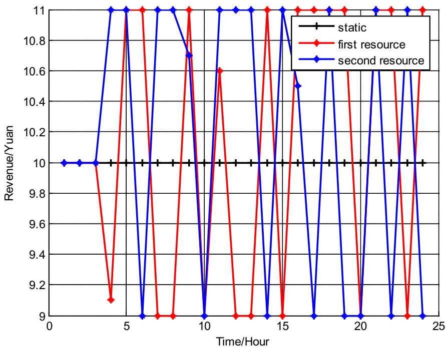

Fig. 4. Price difference between the dynamic model using k-means method and the static model

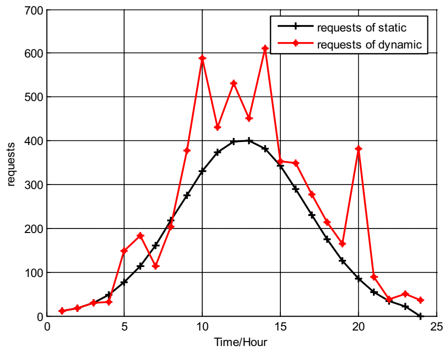

Fig. 5. Requests amount difference between the dynamic model using k-means method and the static model

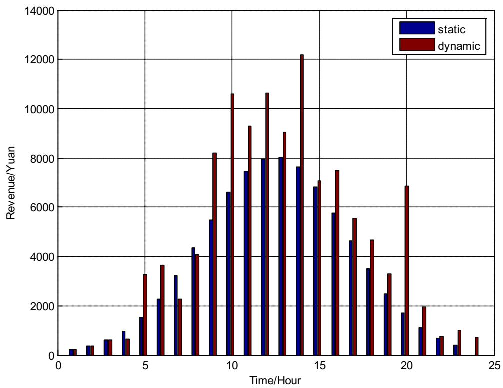

Fig. 6. Hourly revenue difference between the dynamic model using k-means method and the static model

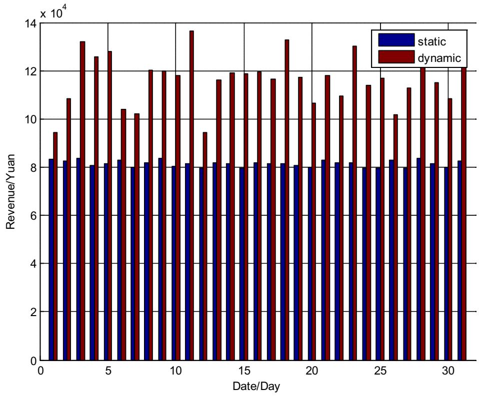

Fig. 7. Daily revenue difference between the dynamic model using k-means method and the static model

4.2 Performance of K-Means in the Algorithm

In this section, a comparison is made that whether there is a classified the data based on different groups of users. In Fig.5 it can be seen if a c-means to classify the users' behaviors into different groups is applied, more revenue can be gotten.

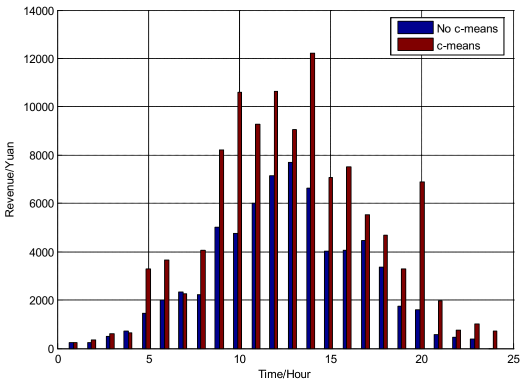

Fig. 8. Hourly revenue difference between the model using k-means method and the model without using k-means method

Acknowledgement. This paper is supported by Doctoral Fund of Ministry of Education of China (20123108120027), by Science and Technology Commission of Shanghai Municipality (No. 10510500600 and No. 12511502900), by State Key Laboratory of Software Engineering(SKLSE) SKLSE2012-09-36, by State Key Laboratory of Novel Software Technology KFKT2012B30 and by Shanghai Leading Academic Discipline Project (No. J50103)。

References

1. Peng, B., Cui, B., Li, X.: Implementation Issues of a Cloud Computing Platform. Bulletin of the Technical Committee on Data Engineering 32(1), 59–67 (2009)
2. Xu, H., Li, B.: Maximizing revenue with dynamic cloud pricing: The infinite horizon case. In: Proc. of IEEE ICC, Next-Generation Networking Symposium, pp. 2929–2933. IEEE Press, Budapest (2012)
3. Caron, E., Desprez, F., Muresan, A.: Pattern Matching Based Forecast of Non-periodic Repetitive Behavior for Cloud Clients. J. Grid Computing, 49–64 (2011)
4. Foster, I., Zhao, Y., Raicu, I., Lu, S.: Cloud Computing and Grid Computing 360-degree compared. In: Grid Computing Environments Workshop, Austin, pp. 1–10 (2008)
5. Constantinos, E., Hill, N.: Cloud Computing for Parallel Scientific HPC Applications: Feasibility of Running Coupled Atmosphere-Ocean Climate Models on Amazon's EC2. In: Cloud Computing and Its Applications, Chicago, IL (2008)
6. Kondo, D., Javadi, B., Malecot, P., Cappello, F., Anderson, D.: Cost-Benefit Analysis of Cloud Computing versus Desktop Grids. In: IEEE International Symposium on Parallel & Distributed Processing, pp. 1–12. IEEE Press, Rome (2009)
7. Anandasivam, A., Premm, M.: Bid Price Control and Dynamic Pricing in Clouds. In: 17th European Conference on Information Systems, Verona, pp. 328–341 (2009)
8. Wang, D., Zeng, X., Keane, J.: A clustering algorithm for radial basis function neural network initialization. Neurocomputing 77(1), 144–155 (2012)
9. Rathnayake, V., Premaratne, H., Sonnadara, D.: Performance of neural networks in forecasting short range occurrence of rainfall. Journal of the National Science Foundation of Sri Lanka 39(3), 251–260 (2011)
10. Feng, Y., David, H.: A short-range quantitative precipitation forecast algorithm using back-propagation neural network approach. Advances in Atmospheric Sciences 23(3), 405–414 (2006)
11. Luis, R., Eddy, C., Adrian, M., Frédéric, D.: Using clouds to scale grid resources: An economic model. Future Generation Computer Systems 28, 633–646 (2012)
12. Yang, X., Xie, W., Huang, J.: A c-means clustering approach based on cloud model. In: IEEE International Conference on Fuzzy Systems, pp. 965–968. IEEE Press, Hong Kong (2008)