International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.5, No.9, September 2016 DOI:10.15693/ijaist/2016.v5i9.18-27

Prediction based Survival Rate for Provisioning Cost in Cloud Computing

Ranjan Kumar
Assistant Professor/ Department of CSE
Cambridge Institute of Technology, Ranchi, India

Dr. G. Sahoo
Professor/Department of CSE
Birla Institute of Technology, Ranchi, India

***Abstract*— Cloud Computing and its different services is a new paradigm in IT sector. Our all data are in cloud and they are very much secure. Only encryption and decryption operation is done by client, all computation process is done in cloud. In this paper we will create some sample data like name, age, health status etc for building relationship on different factors which will be important for life expectancy.**

Index terms - Cloud Computing; Data Set; Provisioning Cost; Sample Data

I. INTRODUCTION

As we all know that Cloud Computing is on demand service. We have to pay as per our use. Cloud storage may be categorized into two major classes: managed and unmanaged storage. Managed storage is provisioned and provides as a raw disk. It depends upon the user to partition and format the disk, attach or mount the disk, and make the storage assets available to applications and other users. In unmanaged, the storage service provider makes storage capacity available to the users. The unmanaged storage is more reliable, cheap to use and also easy to work. The cloud storage domain model is mentioned in figure 1.

The cloud is divided into two parts, the below part is infrastructure and the above part is resource. The infrastructure part consists of facilities, hardware, abstraction, core connectivity & delivery and APIs. The resource part consist of integration & middleware, data, metadata, content, applications, APIs, presentation mobility and presentation platform. The cloud reference model is mentioned below in figure 2. The IaaS consist of purely infrastructure, it is not consist of any of the resources, whereas PaaS have a resource APIs which do management. And the SaaS is in both infrastructure as well as resources. Our data is a part of resources, the structure and unstructured are the part of data, metadata and content. Whereas data, video, voice etc are part of presentation mobility and presentation platform.

In this paper, our concern is for data only. And for this we are using R language to simulate the data sets. Suppose we have large gathering of 800 people in an auditorium from different age group of people. All these data are stored safely in cloud storage. All the people have been classified on the basis of age group, gender, ticket type. We have divided the tickets in 3 classes: Class 1, Class 2 and Class 3. Each attendee is given some customer id associated to the ticket he/she had bought. We have again done fine classification based on the age group and initials whether a person is an adult or kid and is being accompanied by sibling/spouse/parent. One more parameter has been taken which talks about health status. Suppose a stampede happens and 400 people are found to be lost after that. We are trying to create a ML model which will predict the survival of different people. We will create a conclusion from database based on some relations. We will implement forest algorithm for predicting the survival rate. Considering all these parameters, we are trying to create sampled data which will used given as an input to this model. This will be done by building relationship on different factors which will be important for life expectancy (Name, Age, Health Status, etc).

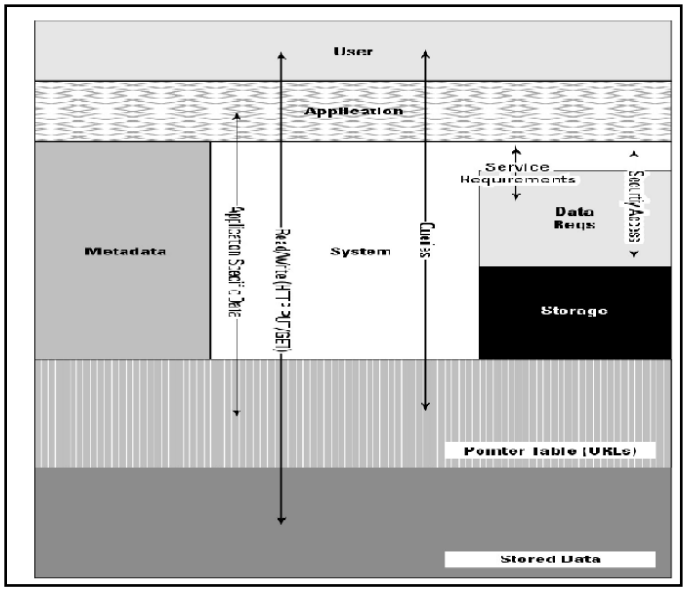


Figure 1. Cloud Storage Domain Model

S No, Life Expectancy, Customer Id, Name, Age, Gender, Title/Initials, Parents/Children, Siblings/Spouse, Class, Ticket Charge.

Here we have made a hypothesis that there is a high chance for survival of rich people then poor people. Among the training data and test data, we will create the distribution among males and females. Last input would be ticket class bought by people. On these 3 inputs, we will create 3D relationship for generating the coefficient id. Another important point would to assess, where there any relatives who were part of big

gathering. This will be done on the basis of name, age and title.

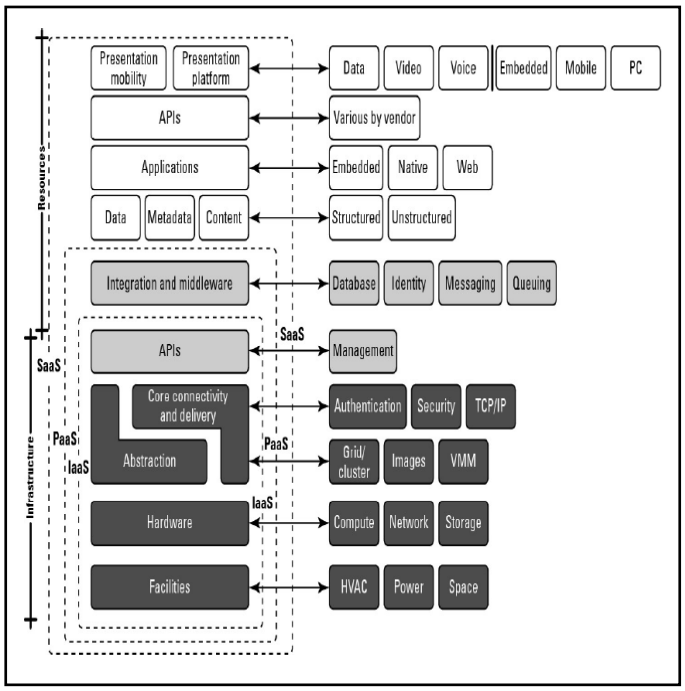


Figure 1. Cloud Computing Reference Model

II. Related Work

Vikram Dayal [2] describes about the graphing, simulating and computing skills to enable one to see theoretical and statistical models in economics in a unified way. It covers some mathematical topics such as, graphing the Cobb-Douglas function, using R to study the Solow growth model, in addition to statistical topics, from drawing statistical graphs to doing linear and logistic regression. It uses data that can be downloaded from the internet, and which is also available in different R packages. With some treatment of basic econometrics, the book discusses quantitative economics broadly and simply, looking at models in the light of data.

Damon M. Berridge [5] describes about Multivariate Generalized Linear Mixed Models Using R presents robust and methodologically sound models for analyzing large and complex data sets.The book applies the principles of modeling to longitudinal data from panel and related studies via the Sabre software package in R. The authors first discuss members of the family of generalized linear models, gradually adding complexity to the modeling framework by incorporating random effects. After reviewing the generalized linear model notation, they illustrate a range of random effects models, including three-level, multivariate, endpoint, event history, and state dependence models. They estimate the multivariate generalized linear mixed models (MGLMMs) using either standard or adaptive Gaussian quadrature. The authors also compare two-level fixed and random effects linear models. The appendices contain additional information on quadrature, model estimation, and endogenous variables, along with SabreR commands and examples. Focusing on these sophisticated data analysis techniques, this book explains the statistical theory and modeling involved in longitudinal studies.

Roman Ilin [7] was proposed new methodology for simultaneous model selection and parameter estimation for a mixture of high-dimensional binary vectors. They found that accelerated MAP is able to correctly identify the number of components in data sets, as long as the overlap of these components is small enough. Accelerated MAP demonstrated superior performance when compared to two state-of-the-art clustering algorithms. The algorithm performed well on both synthetic and real-world data. Using considerations in the guidelines for algorithm's parameter selection were introduced and verified experimentally.

Ian J. Goodfellow, Aaron Courville and Yoshua Bengio [8] motivated the use of the S3C model for unsupervised feature discovery. We have described a variational approximation scheme that makes it feasible to perform learning and inference in large-scale S3C and PDDBM models. They have demonstrated that S3C is an effective feature discovery algorithm for both supervised and semi-supervised learning with small amounts of labeled data. That work addresses two scaling problems: the computation problem of scaling spike-and-slab SC to the problem sizes used in object recognition, and the problem of scaling object recognition techniques to work with more classes. We demonstrate that this work can be extended to a deep architecture using a similar inference procedure, and show that the deeper architecture is better able to model the input distribution. Remarkably, this deep architecture does not require greedy training, unlike its DBM predecessor.

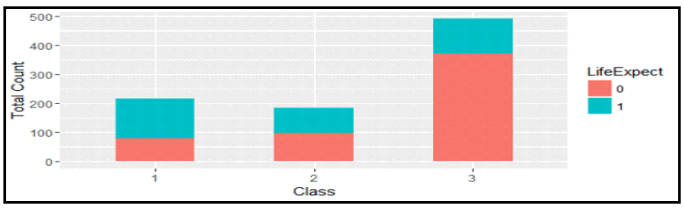
Qingchen Zhang, Laurence T. Yang, and Zhikui Chen [10] was described about Big data offers the great opportunities and transformative potential for various areas such as e-commerce, healthcare industry manufacturing, social network and educational services. Therefore, deep computation, a novel area, has attracted great interests of researchers in recent years. It refers to a systematical model for big data representation, storage, analytic and mining based on tensor theory. They proposed a privacy preserving deep learning model for big data feature learning by incorporating the computing power of the cloud. The proposed scheme uses the BGV encryption scheme to support the secure computation operations of the high-order back-propagation algorithm efficiently for deep computation model training on the cloud. They described that only the encryption operations and the decryption operations are performed by the client while all the computation tasks are performed on the cloud.

III. Objectives & Overview Of The Proposed MECHANISM

Our methodology consists of seven steps which will show the entire work. Assuming Training data is 891 records with 10 columns (same as test data except LifeExpect) and Test data is Around 400 records with 10 columns. We have made some assumptions during the implementation:

Step 1

There is a hypothesis -that Rich folks survived at a higher rate.



On analyzing the above graph, one can conclude that after any emergency, how many people survived from a gathering from among classes.

Class I: Out of 200 people, 120 survived. Survival rate counts to 60%.

Class II: Out of 180 people, 80 survived, Survival rate counts to 44%.

Class III: Out of 480 people, 100 survived, Survival rate counts to 48%.

Step 2:

As mentioned above that we have a training data set of 891 records, out of which we will consider few records for observing names. Some of the data is mentioned below for reference purpose

```
> head(as.character(Training_Data$Name))
[1] "Braund, Mr. Owen Harris"
[2] "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
[3] "Heikkinen, Miss. Laina"
[4] "Futrelle, Mrs. Jacques Heath (Lily May Peel)"
[5] "Allen, Mr. William Henry"
[6] "Moran, Mr. James"
>
```

Based on the above information , we need to identify the unique names across both training data & test data. This will form the basis for identifying the relationship between different attendees in that gathering. If unique names are found, then we can consider that these people were not accompanied by anyone.

```
> length(unique(as.character(data.combined$Name)))
[1] 1307
>
```

Step 3:

Next step is to check for duplicate names, if any:

Incase two duplicate names are found, then take a closer look at other parameters too, as this will form the basis for establishing relationship among attendees based on their accompanied person.

First get the duplicate names and store them as a vector. Then check the gender and see how many males or females are there and their marital status, also check their age. After checking these details, we can establish a relationship among family members (son, daughter, husband, wife, grandparents, etc)

Misses=data.combined[which(str_detect(data.combined$Name, "Miss.")),]

misses [1:5,]

```
>
> misses = data.combined[which(str_detect(data.combined$Name, "Miss.")),]
> misses[1:5,]
   SNo LifeExpect      Customerid                                Name Age Gender ParCh SibSp Class
3    3         1 STON/O2. 3101282                   Heikkinen, Miss. Laina  26 female     0     0     3
11  11         1        PP 9549          Sandstrom, Miss. Marguerite Rut   4 female     1     1     3
12  12         1         113783                 Bonnell, Miss. Elizabeth  58 female     0     0     1
15  15         0         350406 Vestrom, Miss. Hulda Amanda Adolfina  14 female     0     0     3
23  23         1         330923             McGowan, Miss. Anna "Annie"  15 female     0     0     3
     Fare
3  7.9250
11 16.7000
12 26.5500
15  7.8542
23  8.0292
```

Step 4:

Then we have made one more hypothesis - that name titles correlates with age. For example, if the title is "Mrs" that confirms the gender is female and age is above 18 years. Similarly for other cases, in case title is "Master", then gender is male and age is less then 15 years.

Misses=data.combined[which(str_detect(data.combined$Name, "Mrs.")),]

misses [1:5,]

```
> misses = data.combined[which(str_detect(data.combined$Name, "Mrs.")),]
> misses[1:5,]
   SNo LifeExpect Customerid                                        Name Age Gender ParCh
2    2         1   PC 17599 Cumings, Mrs. John Bradley (Florence Briggs Thayer)  38 female     0
4    4         1     113803        Futrelle, Mrs. Jacques Heath (Lily May Peel)  35 female     0
9    9         1     347742   Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  27 female     2
10  10         1     237736                 Nasser, Mrs. Nicholas (Adele Achem)  14 female     0
16  16         1     248706                    Hewlett, Mrs. (Mary D Kingcome)   55 female     0
   SibSp Class    Fare
2      1     1 71.2833
4      1     1 53.1000
9      0     3 11.1333
10     1     2 30.0708
16     0     2 16.0000
```

Similarly we need to check out males to see, if the same pattern continues

Males = data.combined[which(train$Gender == 'male'),]

males [1:5,]

```
> males = data.combined[which(Training_Data$Gender == 'male'),]
> males[1:5,]
  SNo LifeExpect Customerid                      Name Age Gender ParCh SibSp Class   Fare
1   1         0  A/5 21171       Braund, Mr. Owen Harris  22   male     0     1     3 7.2500
5   5         0     373450      Allen, Mr. William Henry  35   male     0     0     3 8.0500
6   6         0     330877             Moran, Mr. James  NA   male     0     0     3 8.4583
7   7         0      17463       McCarthy, Mr. Timothy J  54   male     0     0     1 51.8625
8   8         0     349909 Palsson, Master. Gosta Leonard  2   male     1     3     3 21.0750
>
```

After getting the count for male and females and their age group then we plan to expand the relationship between 'LifeExpect' and 'Class' by adding the new variable called "Title"

Using this data set we explore a potential multi-dimensional relationship.

Now next step is how to extract the title for all people from varied age group.

This will be done by creating a utility function for title extraction extractTitle = function (name) {

```
 name = as.character(name)
  if(length(grep("Miss.", name)) > 0){
  return ("Miss.")
 }else if(length(grep("Master.", name)) > 0){
  return ("Master.")
 }else if(length(grep("Mrs.", name)) > 0){
  return ("Mrs.")
 }else if(length(grep("Mr.", name)) > 0){
  return ("Mr.")
 }else{
  return ("Other")
 }

}
titles = NULL
for(i in 1:nrow(data.combined)){
 titles = c(titles, extractTitle(data.combined[i, "Name"]))
}
data.combined$Title = as.factor(titles)
```

Expect	Customerid	Name	Age	Gender	ParCh	SibSp	Class	Fare	Title
	A/5 21171	Braund, Mr. Owen Harris	22.00	male	0	1	3	7.2500	Mr.
	PC 17599	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	38.00	female	0	1	1	71.2833	Mrs.
	STON/O2. 3101282	Heikkinen, Miss. Laina	26.00	female	0	0	3	7.9250	Miss.
	113803	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.00	female	0	1	1	53.1000	Mrs.
	373450	Allen, Mr. William Henry	35.00	male	0	0	3	8.0500	Mr.
	330877	Moran, Mr. James	NA	male	0	0	3	8.4583	Mr.
	17463	McCarthy, Mr. Timothy J	54.00	male	0	0	1	51.8625	Mr.
	349909	Palsson, Master. Gosta Leonard	2.00	male	1	3	3	21.0750	Master.
	347742	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	27.00	female	2	0	3	11.1333	Mrs.
	237736	Nasser, Mrs. Nicholas (Adele Achem)	14.00	female	0	1	2	30.0708	Mrs.

Step 5:

Now we have identified correlation among different entities and relationship is clearly evident. Next step is to generate LifeExpect lables for the training data set:

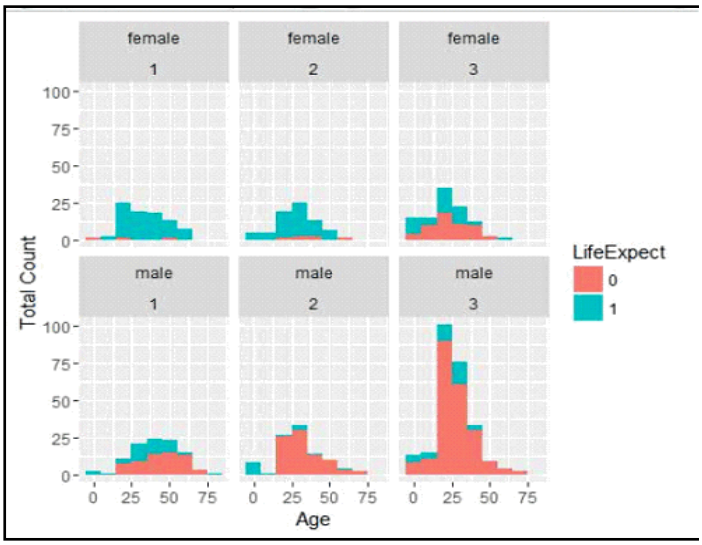
```
ggplot(data.combined[1:891,], aes(x = Title, fill = Survived)) +
  geom_bar(binwith = 0.5) +
  facet_wrap(~Pclass) +
  ggtitle("Pclass") +
  xlab("Title") +
  ylab("Total Count") +
  labs(fill = "Survived")
```

Step 6: Next step is to plot distribution of age group and gender type in mob:

```
> summary(data.combined$Age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.17   21.00   28.00   29.88   39.00   80.00     263
> summary(data.combined[1:891,"Age"])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.42   20.12   28.00   29.70   38.00   80.00     177
```

Take a look at survival rates based on gender, pclass and age

```
> ggplot(data.combined[1:891,], aes(x = Age, fill = LifeExpect)) +
+   facet_wrap(~Gender + Class) +
+   geom_histogram(binwidth = 10) +
+   xlab("Age") +
+   ylab("Total Count")
```



We generally assume that "Master" is a good proxy for male children depending on his age:

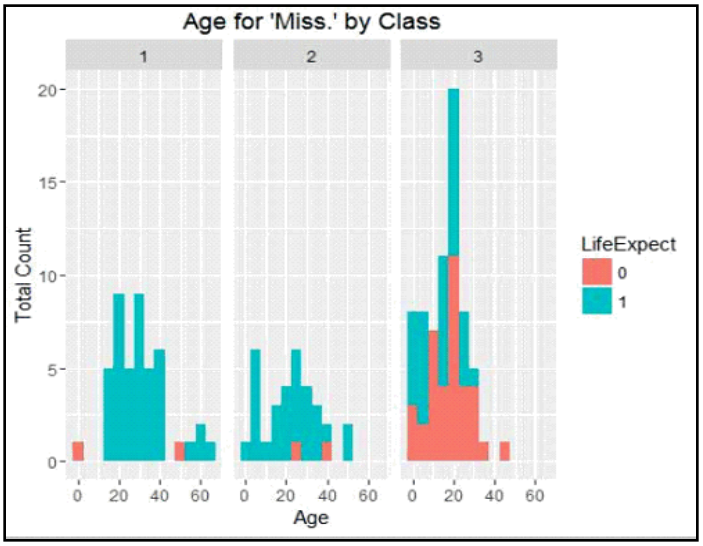
```
> summary(boys$Age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  0.330   2.000   4.000   5.483   9.000  14.500       8
```

We know that "Miss." is more complicated, let's examine further (can include females of different age group)

```
> misses <- data.combined[which(data.combined$title == "Miss."),]
> summary(misses$Age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.17   15.00   22.00   21.77   30.00   63.00      50
```

Here we have plotted survival distribution for miss in different classes

```
> ggplot(misses[misses$LifeExpect != "None",], aes(x = Age, fill = LifeExpect)) +
+   facet_wrap(~Class) +
+   geom_histogram(binwidth = 5) +
+   ggtitle("Age for 'Miss.' by Class") +
+   xlab("Age") +
+   ylab("Total Count")
```



Female children may have different survival rate, this could be a candidate for feature engineering later

```
> misses.alone <- misses[which(misses$SibSp == 0 & misses$ParCh == 0),]
> summary(misses.alone$Age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   5.00   21.00   26.00   27.23   32.50   58.00      33
> length(which(misses.alone$Age <= 14.5))
[1] 4
```

Move on to the SibSp variable, summarize the variable

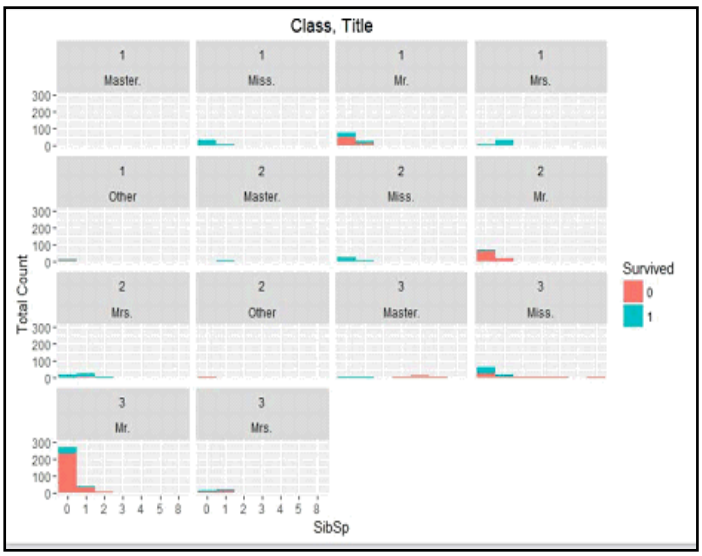
```
> summary(data.combined$SibSp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0000  0.0000  0.4989  1.0000  8.0000
```

Can we treat as a factor?

```
> length(unique(data.combined$SibSp))
[1] 7
```

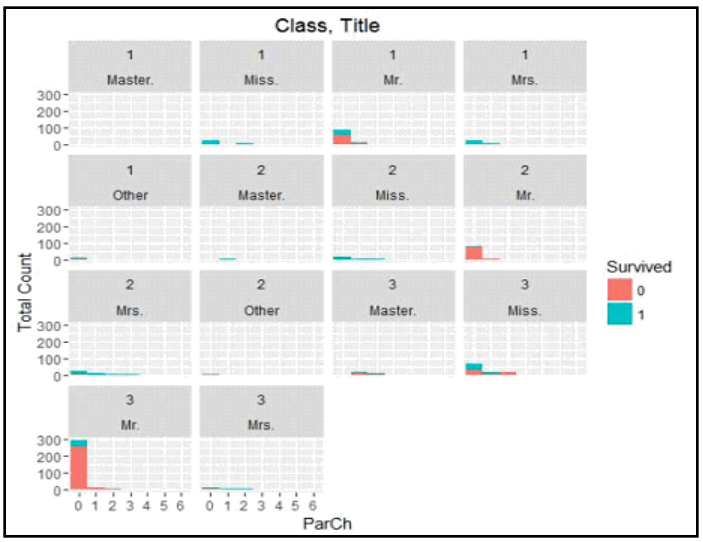
We believe that title is predictive. Now visualize survival rates by SibSp, pclass, and Title.

```
> data.combined$SibSp <- as.factor(data.combined$SibSp)
> ggplot(data.combined[1:891,], aes(x = SibSp, fill = LifeExpect)) +
+   stat_count(width = 1) +
+   facet_wrap(~Class + title) +
+   ggtitle("Class, Title") +
+   xlab("SibSp") +
+   ylab("Total Count") +
+   ylim(0,300) +
+   labs(fill = "Survived")
```



Now considering the ParCh variable as a factor and visualize

```
> data.combined$ParCh <- as.factor(data.combined$ParCh)
> ggplot(data.combined[1:891,], aes(x = ParCh, fill = LifeExpect)) +
+   stat_count(width = 1) +
+   facet_wrap(~Class + title) +
+   ggtitle("Class, Title") +
+   xlab("ParCh") +
+   ylab("Total Count") +
+   ylim(0,300) +
+   labs(fill = "Survived")
```

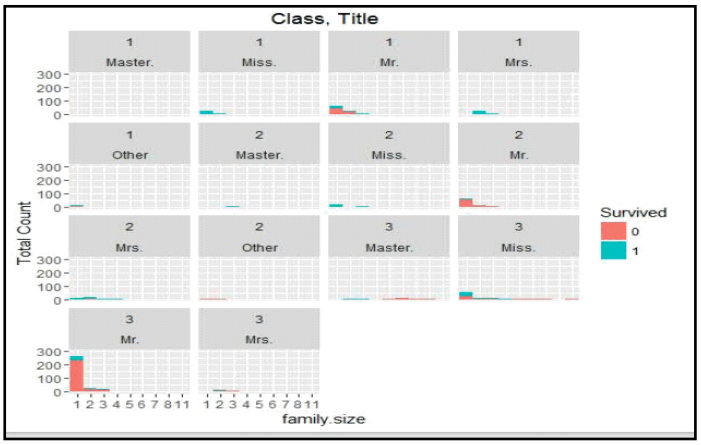


Let's try some feature engineering. Lets try to create a family size feature?
temp.sibsp<-c(train$sibsp,test$sibsp)
temp.parch <- c(train$parch, test$parch)

```
> data.combined$family.size <- as.factor(temp.SibSp + temp.ParCh + 1)
> View(data.combined)
> View(data.combined)
```

	SNo	LifeExpect	Customerid	Name	Age	Gender	ParCh	SibSp	Class	Fare	title	family.size
1	1	0	A/5 21171	Braund, Mr. Owen Harris	22.00	male	0	1	3	7.2500	Mr.	2
2	2	1	PC 17599	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	38.00	female	0	1	1	71.2833	Mrs.	2
3	3	1	STON/O2. 3101282	Heikkinen, Miss. Laina	26.00	female	0	0	3	7.9250	Miss.	1
4	4	1	113803	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.00	female	0	1	1	53.1000	Mrs.	2
5	5	0	373450	Allen, Mr. William Henry	35.00	male	0	0	3	8.0500	Mr.	1
6	6	0	330877	Moran, Mr. James	NA	male	0	0	3	8.4583	Mr.	1
7	7	0	17463	McCarthy, Mr. Timothy J	54.00	male	0	0	1	51.8625	Mr.	1
8	8	0	349909	Palsson, Master. Gosta Leonard	2.00	male	1	3	3	21.0750	Master.	5
9	9	1	347742	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	27.00	female	2	0	3	11.1333	Mrs.	3
10	10	1	237736	Nasser, Mrs. Nicholas (Adele Achem)	14.00	female	0	1	2	30.0708	Mrs.	2
11	11	1	PP 9549	Sandstrom, Miss. Marguerite Rut	4.00	female	1	1	3	16.7000	Miss.	3
12	12	1	113783	Bonnell, Miss. Elizabeth	58.00	female	0	0	1	26.5500	Miss.	1
13	13	0	A/5 2151	Saundercock, Mr. William Henry	20.00	male	0	0	3	8.0500	Mr.	1

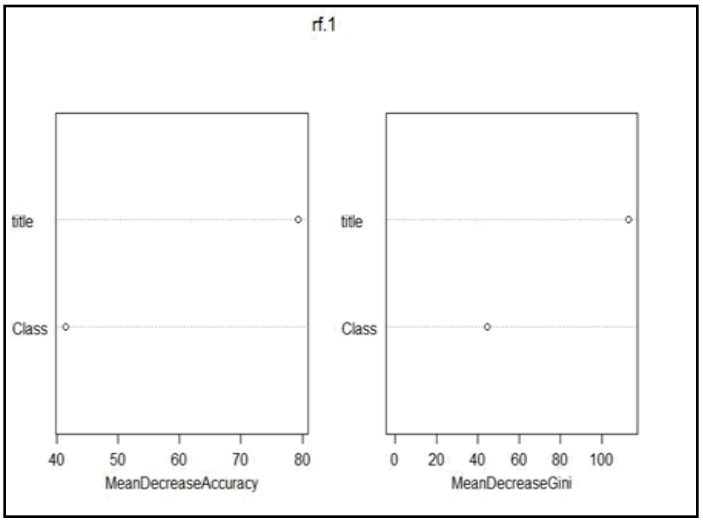
```
> ggplot(data.combined[1:891,], aes(x = family.size, fill = LifeExpect)) +
+   stat_count(width = 1) +
+   facet_wrap(~Class + title) +
+   ggtitle("Class, Title") +
+   xlab("family.size") +
+   ylab("Total Count") +
+   ylim(0,300) +
+   labs(fill = "Survived")
>
```



Step 7:
Train a Random Forest with the default parameters using pclass & Title

First iteration of random forest:

Variables used: Class and title

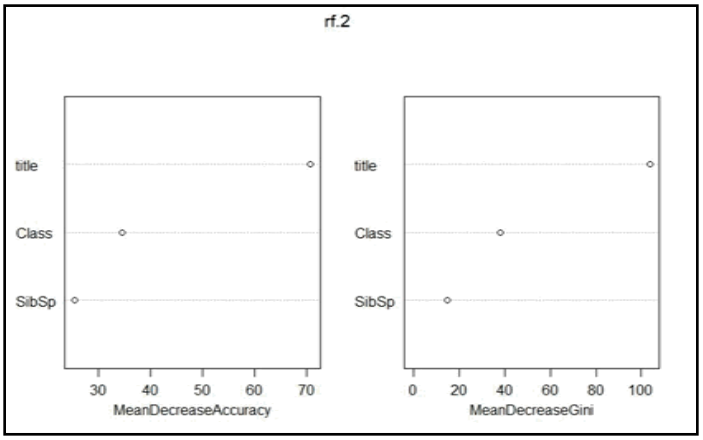


Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 20.76%
Confusion matrix:
    0   1 class.error
0 538  11  0.02003643
1 174 168  0.50877193
> varImpPlot(rf.1)
```

Second iteration of random forest:

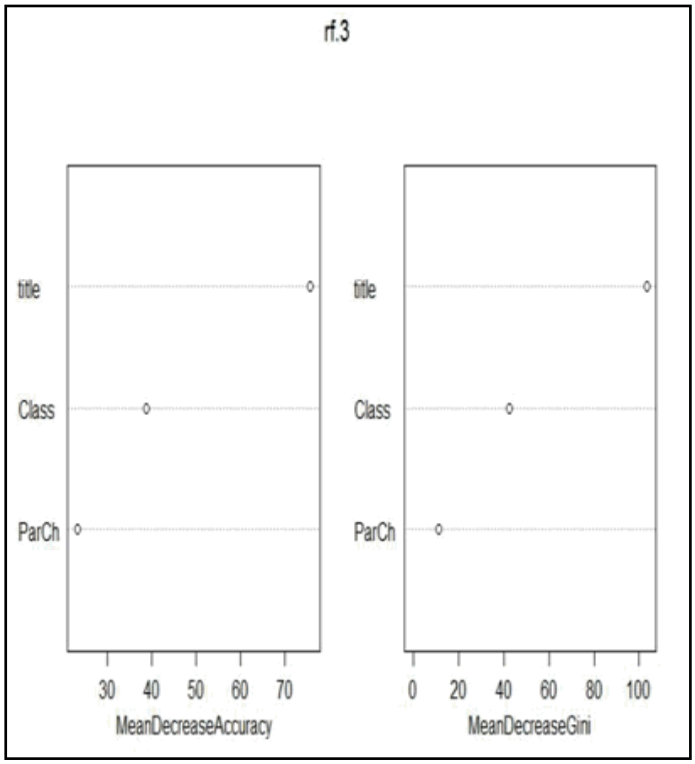
Variables used: Class, SibSp, title



Here we have the error estimate and confusion matrix:

```
        OOB estimate of  error rate: 19.75%
Confusion matrix:
    0   1 class.error
0 487  62   0.1129326
1 114 228   0.3333333
> varImpPlot(rf.2)
```

Third iteration of random forest:
Variables used: Class, title, ParCh

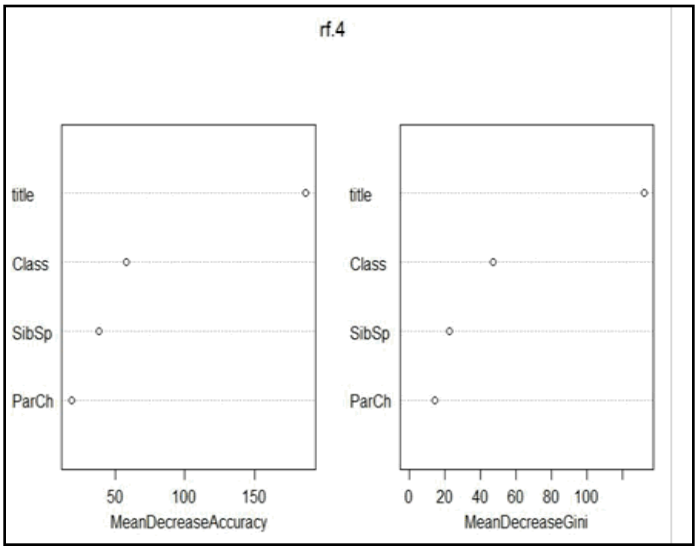


Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 19.98%
Confusion matrix:
    0   1 class.error
0 495  54  0.09836066
1 124 218  0.36257310
> varImpPlot(rf.3)
```

Fourth iteration of random forest:

Variables used: Class, title, SibSp, ParCh

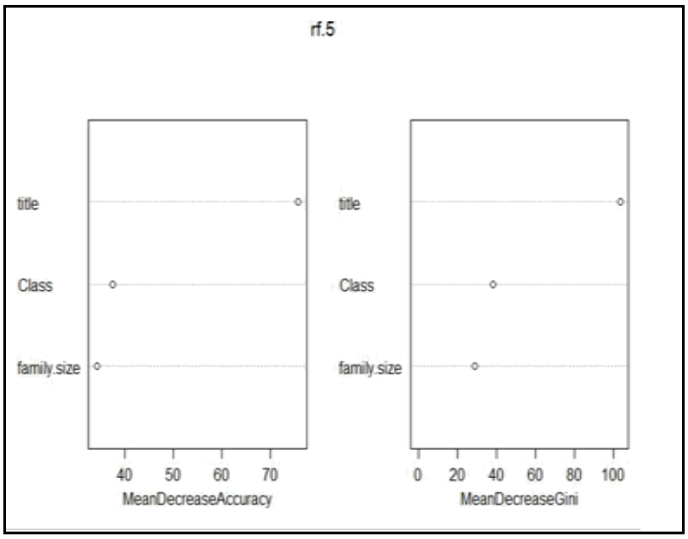


Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 18.52%
Confusion matrix:
    0   1 class.error
0 489  60   0.1092896
1 105 237   0.3070175
> varImpPlot(rf.4)
```

Fifth iteration of random forest:

Variables used: Class, family Size and title

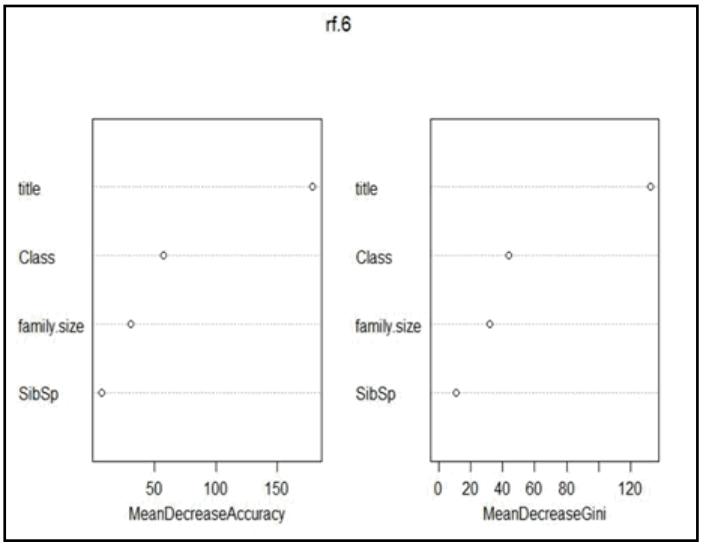


Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 18.41%
Confusion matrix:
    0   1 class.error
0 485  64   0.1165756
1 100 242   0.2923977
> varImpPlot(rf.5)
```

Sixth iteration of random forest:

Variables used: Class, title, family Size, SibSp

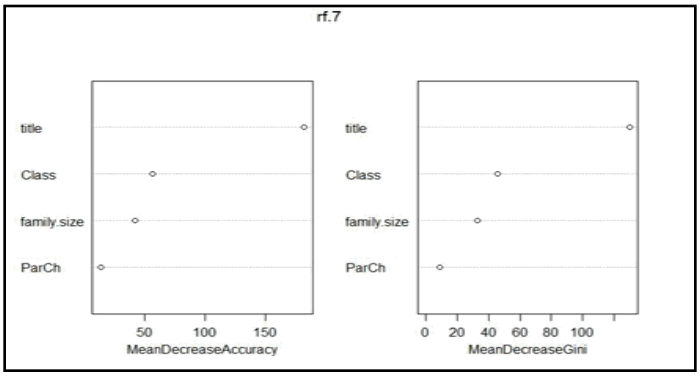


Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 18.74%
Confusion matrix:
    0   1 class.error
0 486  63   0.1147541
1 104 238   0.3040936
> varImpPlot(rf.6)
>
```

Seventh iteration of random forest:

Variables used: Class, title, family Size, ParCh



Here we have the error estimate and confusion matrix

```
        OOB estimate of  error rate: 18.97%
Confusion matrix:
    0   1 class.error
0 486  63   0.1147541
1 106 236   0.3099415
> varImpPlot(rf.7)
>
```

Here is the final prediction survival:

	PassengerId	Survived
892	892	0
893	893	1
894	894	0
895	895	0
896	896	1
897	897	0
898	898	1
899	899	0
900	900	1
901	901	0
902	902	0
903	903	0
904	904	1
905	905	0

V. Performance Evaluation

A. Simulation Resut

We use R language and random forest algorithm to simulate our research. The snapshots of our result are given below.

VI. Conclusion

In this research work, our proposed methodology gives the survival and non-survival rate of the passengers from the data set. It also helps in the resource provisioning of cost in cloud computing. It gives the results for both success rate as well as for error rate.

References

[1]. Lawrence Leemis. *Learning Base R*. Lightning Source, 2016. ISBN 978-0-9829174-8-0. [bib | http://www.amazon.com/Learning-Base-Lawrence-Mark-Leemis/dp/0982917481]

[2]. Vikram Dayal. *An Introduction to R for Quantitative Economics: Graphing, Simulating and Computing*. Springer, 2015. ISBN 978-81-322-2340-5. [bib | http://www.springer.com/978-81-322-2340-5]

[3]. C. Sun. *Empirical Research in Economics: Growing up with R*. Pine Square, Starkville, Mississippi, USA, 1st edition, 2015. ISBN 978-0-9965854-0-8. Supplementary materials are available at http://csun.cfr.msstate.edu. [bib | http://www.amazon.com/Empirical-Research-Economics-Changyou-Sun/dp/0996585400/ref=aag_m_pw_dp?ie=UTF8&m=A1TZL30UWYSSR8]

[4]. Matthias Kohl. *Introduction to statistical data analysis with R*. bookboon.com, London, 2015. ISBN 978-87-403-1123-5. [bib |Publisher Info]

[5]. Damon M. Berridge. *Multivariate Generalized Linear Mixed Models Using R*. Chapman & Hall/CRC Press, Boca Raton, FL, 2011. ISBN 978-1-4398-1326-3. [bib |Discount Info | http://www.crcpress.com/product/isbn/9781439813263]

[6]. Barrie Sosinsky, "Cloud Computing Bible", Wiley Publishing Inc, 2011.

[7]. Roman Ilin, "Unsupervised Learning of Categorical Data With Competing Models", IEEE Transaction on Neural Networks and Learning Systems, Vol. 23, No. 11, November, Pp. 1726-1737, 2012.

[8]. Ian J. Goodfellow, Aaron Courville and Yoshua Bengio, "Scaling Up Spike-and-Slab Models for Unsupervised Feature Learning", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 35, No.8, August, Pp. 1902-1914, 2013.

[9].] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi and Fei-Yue, "Traffic Flow Prediction With Big Data: A Deep Learning Approach" IEEE Transaction Transportation System, Vol. 16, No. 2, April, Pp. 865-873, 2015.

[10]. Qingchen Zhang, Laurence T. Yang, and Zhikui Chen, "Privacy Preserving Deep Computation Model on Cloud for Big Data Feature Learning", IEEE Transactions on Computers, vol. 65, no. 5, May, Pp. 1351-1362, 2016.

Authors Profile

Ranjan Kumar received the **B.Tech and M.Tech** degree in computer science & engineering from the Cambridge Institute of Technology and Birla Institute of Technology, Mesra, Ranchi, India. Currently doing **Ph.D** in computer science & engineering (Cloud Computing) from Birla Institute of Technology, Mesra, India. His research interest includes Cloud Computing, Grid Computing.

Dr. G. Sahoo received the **Ph.D** degree from IIT Kharagpur, India. He is Professor in Department of Computer Science & E ngineering and Dean of Admissions & Academic Coordination in Birla Institute of Technology, Mesra, Ranchi. He has more than 250 research publications. His research interest includes Cloud Computing, Theoretical Computer Science, Sequential & Parallel Computing, Pattern Recognition, Grid Computing, Data Mining, Bio-Informatics, Cryptography and Data Security.