

## **PREDICTION OF HOSPITAL ADMISSION USING MACHINE LEARNING**

**KAVYA AMBATI<sup>1</sup>, GEETHIKA REDDY GUNDA<sup>2</sup>, JISHITHA SURAKANTI<sup>3</sup>,  
KALYANI LOTLAPALLI<sup>4</sup>, KAVYA SRIRANGAM<sup>5</sup>, Y LAXMI PRASANNA<sup>6</sup>**

<sup>1,2,3,4</sup>UG students, Dept of CSE, ANURAG Engineering College, Ananthagiri, Suryapet, TS, India.

<sup>5</sup>Assistant Professor, Dept of CSE, ANURAG Engineering College, Ananthagiri, Suryapet, TS, India.

### **ABSTRACT:**

People will face many problems in Hospitals while taking Admission. If it is in a popular hospital, they should wait hours together to take just admission. But it is not at all good at Emergency Department. Very serious cases will admit in Emergency Department. So, we need to use more innovation technique to ameliorate patient flow and prevent Overflowing. So, data mining techniques will show us a pleasant method to predict the ED Admissions. Here we Analyzed an algorithm for predicting models i.e Naive Bayes, Random Forests, Support Vector Machine. For the prediction we should identify a handful of factors associated to Hospital admission including age, gender, systolic pressure, diastolic pressure, diabetes, previous records in the preceding month or year, admission. We also say about the algorithms which we used in detail. We use Random Forests algorithm for classifying the data into categories for improving the accuracy of prediction. Naive Bayes is used to identify the probabilities for each attribute and helps in predicting the outcome. Support Vector machine is used to classify the given input particular category which helps in predicting the outcome.

**Keywords:** *ML, Random forest, SVM, Naive bayes, high accuracy.*

### **1. INTRODUCTION**

One of the biggest yet overlooked problems in the Medical Industry is Emergency Department Crowding. These are the most severely injured or patients who need immediate attention. However, it is often very difficult to identify the state of all the patients in the

Emergency ward which leads to making wrong decisions which soon leads to overcrowding. This is why the ability to identify the state of a patient has become crucial worldwide. Overcrowding might seem like an easy problem to get over but in reality it is very hard to handle. The consequences are harsh and will directly impact the patients



as well as the staff in the hospital as the wait times will increase drastically and it will be too late for anyone to react due to the shortage of required staff. This is why it is necessary for us to come up with innovative approaches to solve this global issue to improve the patient flow and preventing patient crowding. One of the best approaches to this method over the past few years has been the use of data mining using various ML techniques in order to predict the state of various emergency patients that are currently admitted in the hospital. However, there are a few cases in which emergency crowding takes place due to the shortage of doctors or even the lack of inpatient beds. These are mainly caused due to the fact that the patients from the emergency ward are transferred to these inpatient beds. This is one of the problems we can easily rectify with the help of data mining in order to identify patients that are inpatient admissions from those who are not so that we can avoid any confusions in our system. In this study we will mainly focus upon implementing various machine learning algorithms and developing models in order to predict the state of the patients that are being admitted into the emergency department. We will also

be comparing the performance of our model with a few various approaches that are already in the world. Patients who plan on visiting the hospitals for various issues and those that are in the emergency department will be required to go through various phases between the time that they arrive to their time of discharge. In These phases will focus upon the various decisions that they had to make depending upon their previous phases. During these phases we will collect various data from the patients such as their patient's age, gender, systolic pressure, diastolic pressure, diabetes, previous records based on these factors the patient will be admitted.

## **2. RELATED STUDY**

The emergency attendees may come through main reception or in ambulance at this point of time depending on the situation of the patient the details should be taken for some of the medications age, gender, blood pressure, diabetes play a vital role for the further treatment. Usually to collect the data from the patient it takes ten to fifteen minutes the patient who comes with emergency may not have time to complete all this procedure. To identify such cases, we must use a Triage Scale in order to understand the condition of the patient and how



urgently they require medical care. This is one of the most important phases for the safety of any patient. When we look into the previous records of any hospital we can clearly identify that there were far more aged people admitted in the hospital when compared to children or adults. This has caused chaos at emergency departments due to a lack of knowledge regarding the procedures and department medical systems. The number of visit rates to a hospital has been rising rapidly over the past decade. Due to this it is essential for us to create a quick and accurate Triage System in order to assess all the patients. Once the patient has undergone the Triage Process they will be shifted to the clinical room where they will be consulted by a clinician who will provide the best course of action for the patient. There are various Triage Systems that are used commonly around the world. However, the two most commonly used triage systems are those that use either a 3 Level Triage System or a 5 Level Triage System. A 3 Level Triage System labels patients as Emergent, Urgent and NonUrgent from the highest to lowest level respectively. Similarly, the 5 Level System is broken down as Resuscitation, Emergent, Urgent, Less Urgent, Not Urgent from

lowest level to highest level respectively. Various studies around the world have showed that the 5 Level Triage System has been far more reliable than the standard 3 Level System. It has done a better job in predicting the consumption of resources, length of stay, admission rates and mortality. Building a Triage System that is highly accurate and precise can play a major impact in the medical industry as it could save millions of lives. Our Study is based upon two major objectives. Our first objective is to create and develop a model that is able to accurately predict whether a patient from the emergency department will be admitted into the hospital. Our Later objective is to study the performance of various other machine learning algorithms in this sector. In order to predict the state of a patient we must first have our heads wrapped around the knowledge of various mathematical models. The previous research was done by using logistic regression, decision tree and time series forecasting algorithms. In the previous analysis when compared with other algorithms like logistic regression, decision tree and gradient boosted. Gradient boosted got the more accurate as we use decision tree it is not suitable for



longer data sets and need to perform pruning in decision trees whereas in gradient boosted it merges the weaker trees and forms the stronger one which helps in the prediction. According to the statistics the rate of patient stays, or visits was gradually increased from the year 2005 to 2014. Annual average growth rate for impatient stay was 5.7% and cumulative increase was 64.1% where as in Emergency Department visits annual average growth rate was 8.0% and cumulative increase was 99.4%. Objective is to find the model which suits the best and gives the accurate results for predicting the admission in the emergency department. Here the comparison of three machine learning algorithms was done (i.e.) Naïve Bayes, Support vector machine (SVM), Random forest classifier. After comparison Support Vector machine got the most accurate results when compared to others.

### **EXISTING SYSTEM**

Random forest is a commonly used tool in the construction of Decision trees. Instead of following the normal routine it takes a subset of variables and observations in order to construct the decision tree. It builds various decision trees and merges them together in order to form a single decision tree that has

high accuracy and prediction. The Random Forest is generally viewed upon as a black box as its predictions are highly accurate. Most people don't bother about the background calculations due to its high accuracy rate. Although we won't be able to change the methods of calculations for the Random Forest it has a few modifiable factors which can in turn effect the performance of the model or the resources and time balance. We will talk about their variable factors further on in the construction of our Rainforest Mode

### **3 PROPOSED SYSTEM**

The Classification of Linear as well as Non-Linear Data can be simply completed with the help of a Support Vector Machine (SVM). Let's take a simple look at how SVM's function or work. SVM's apply nonlinear mapping in order to convert the original training data into training data in higher dimensions. Once we have established this new dimension the model will begin searching for linear optimal separating hyperplanes. The SVM is able to find and separate these hyperplanes with the usage of support vectors and margins. We will look deeper into these concepts later on in our study. However, in the past decade SVM's have been attracting a lot of





attention. SVM's were first introduced into the picture when Vladimir Vapnik along with his colleagues Bernhard Boser & Isabelle Guy decided to write a paper on them in 1992. Although these group of researchers were the first to have written a paper on SVM's the concept has dated back to the 1960s. SVM's follow a rather complicated internal structure and the time to train them is extremely slow. However, putting this con aside, you will be able to expect outputs which are highly accurate and precise. Another key factor to using SVM's is their ability to be prone to overfitting. A commonly used application of SVM's has been numeric or alphanumeric prediction as well as classification. Other applications for SVM's has included areas such as hand written language or digit detection, speaker identification, object detection, and Benchmark time series. SVM's are mostly based upon the concepts of decision planes that have predefined boundaries. A decision plane can simply be defined as a barrier that separates the various objects that belong to different membership classes. Let's try to take a look at this simple schematic example in which objects either belong to the left class or the right class. The line

in the middle acts as the boundary or you can say decision plane which separates the right and left class. All the objects that are situated to the left of this line are known as the left class while all those to the right are classified as the right class. When a new object enters into the scenario it falls upon the boundary line which will then make the classification to either push it left or right into its respective class

### **SCOPE :**

When we sort through large data sets in order to identify various patters and establish relationships it is known as Data Mining. These patterns and Relationships can be further used in order to solve various problems through data analytics. Enterprises are able to make predictions upon future trends with the help of Data Mining tools. We are able to do so by using massive amounts of data in order to identify the various patterns and trends. It typically consists of Data Transformation, Pattern Evaluation, Data Cleaning, Pattern Discovery, Data Integration. and Knowledge Presentation. We use Association rules within data mining by exploring and analyzing the data for various if/then patterns. From here we will use various support and confidence criteria in order to form

various important relationships among data. Support is defined as the number of times a specific query is found within a database, while confidence is the probability that the if/then case is accurate. There are other parameters used within data mining such as Sequence or Path Analysis, Clustering and Forecasting, Classification, and Sequence or Path Analysis. An ordered list of a set of items is known as a Sequence. It is commonly found in any sort of Database.

## Prediction Of Hospital Admission Using Machine Learning

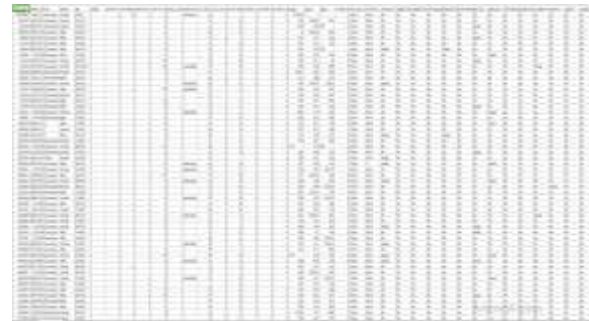
### Data set description:

For hospital admission prediction we take patients data. This data consists fifty columns and 101766 records.

encounter\_id, patient\_nbr, race,gender, age, weight, admission\_type\_id, discharge\_disposition\_id, admission\_source\_id, time\_in\_hospital, payer\_code, m, edical\_specialty, num\_lab\_procedures, num\_procedures, num\_medications, number\_outpatient, number\_emergency, number\_inpatient, diag\_1, diag\_2, diag\_3, number\_diagnoses, max\_glu\_serum, A1Cresult, metformin, repaglinide, nateglinide, chlorpropamide, glimepiride, acetohexamide, glipizide, glyburide, tolbutamide, pioglitazone, rosiglitazone, acarbose, acarbose, miglitol, troglitazone, tolazamide, examide, citoglipton, insulin,

glyburide-metformin, glipizide-metformin, glimepiride-pioglitazone, metformin-rosiglitazone, metformin-pioglitazone, change, diabetesMed, readmitted

Above all names are dataset column names.



### Data Preparation & Exploration

```
loading libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

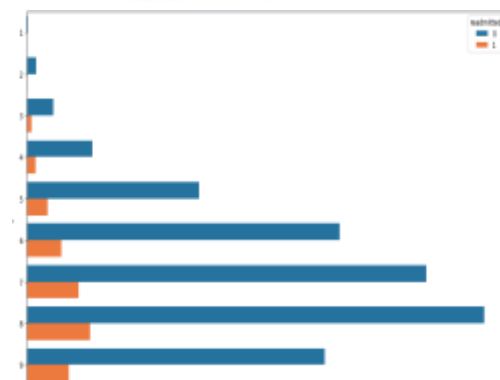
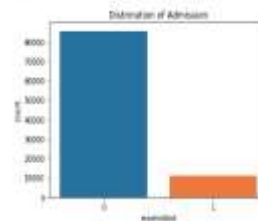
```
loading Dataset
df = pd.read_csv('data/data.csv')
```

```
displaying first 10 rows of data
df.head(10,7)
```

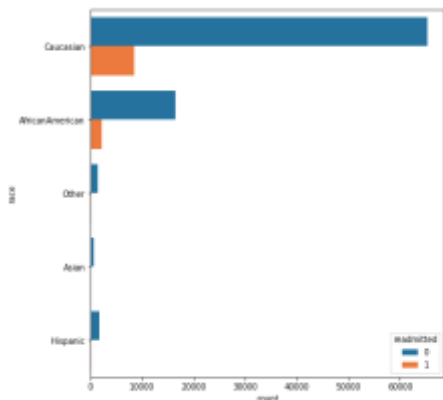
### Data Visualization

```
# Distribution of Readmission
sns.countplot(df['readmitted']).set_title('Distribution of Admission')
/usr/local/lib/python3.4/dist-packages/seaborn/decorators.py:43: FutureWarning: Pass the following variable as keyword arg: x. From version 0.12, the only valid positional argument will be data, and passing other as without an explicit keyword will result in an error or misinterpretation.
FutureWarning
```

Text(0.5, 1.0, 'Distribution of Admission')



## Based on age and admission



```

1: from sklearn.tree import DecisionTreeClassifier
dtree = DecisionTreeClassifier(max_depth=10, criterion = 'entropy', min_samples_split=10)
dtree.fit(X_train, y_train)

1: DecisionTreeClassifier(ccp_alpha=0.0, class_weight='none', criterion='entropy',
max_depth=10, max_features=None, max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=10,
min_weight_fraction_leaf=0.0, presorted=False,
random_state=None, splitter='best')

1: dtree_pred = dtree.predict(X_test)
pd.crosstab(pd.Series(y_test), pd.Series(dtree_pred), name = 'Predict'), margins = True)

1:
Predict  0  1  All
Actual
0  3027  704 13000
1  1234  8794 10010
All 11361 10498 21859

```

## Prediction based on RF

```

From sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators = 10, max_depth=25, criterion = 'gini', min_samples_split=10)
rf.fit(X_train, y_train)

RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
criterion='gini', max_depth=25, max_features='auto',
max_leaf_nodes=None, max_samples=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=10,
min_weight_fraction_leaf=0.0, n_estimators=10,
n_jobs=None, oob_score=False, random_state=None,
verbose=0, warm_start=False)

rf_pred = rf.predict(X_test)
pd.crosstab(pd.Series(y_test), pd.Series(rf_pred), name = 'Predict'), margins = True)

Predict  0  1  All
Actual
0  3276  147 13000
1  1107  8811 10010
All 11383 10000 21383

```

## Based on race and admission

```

df['age'] = df['age'].astype('int64')
print(df.age.value_counts())
# convert age categories to mid-point values
age_dict = {'1:5', 2:15, 3:25, 4:35, 5:45, 6:55, 7:65, 8:75, 9:85, 10:95}
df['age'] = df.age.map(age_dict)
print(df.age.value_counts())

8    24815
7    21521
6    16546
9    16223
5     9288
4     3538
10    2594
3     1471
2         466
1          64
Name: age, dtype: int64
75    24815
65    21521

```

## Prediction based on SVM

```

1: from sklearn.svm import SVC
svm = SVC()
svm.fit(X_train, y_train)
svm_pred = svm.predict(X_test)
pd.crosstab(pd.Series(y_test), pd.Series(svm_pred), name = 'Predict'), margins = True)

Predict  0  1  All
Actual
0  3276  147 13000
1  1107  8811 10010
All 11383 10000 21383

print("Accuracy is %.2f" % svm.score(svm_test, svm_pred))
print("Precision is %.2f" % svm.score(svm_test, svm_pred))
print("Recall is %.2f" % svm.score(svm_test, svm_pred))
svm_acc = svm.score(svm_test, svm_pred)
svm_acc = svm.score(svm_test, svm_pred)
svm_re = svm.score(svm_test, svm_pred)

Accuracy is 0.94
Precision is 0.99
Recall is 0.98

```

## Modelling and Data preprocessing

## ML deploying

```

1: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import log_loss, accuracy_score
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)
logit = LogisticRegression(fit_intercept=True, penalty='l2')
logit.fit(X_train, y_train)

~/usr/local/lib/python3.8/dist-packages/sklearn/linear_model/_logistic.py:444: ConvergenceWarning: lbfgs failed to
converge (13 iterations)
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
https://www.kaggle.com/colsonleung/understanding-ml
Please also refer to the documentation for alternative solver options:
https://www.kaggle.com/colsonleung/understanding-ml#logistic-regression
extra_warning_msg="LOGISTIC SOLVER CONVERGENCE MSG")

1: LogisticRegression(C=0.8, class_weight=None, dual=False, fit_intercept=True,
intercept_scaling=1, l1_ratio=None, max_iter=100,
multi_class='auto', n_jobs=None, penalty='l2',
random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
warm_start=False)

```

## Prediction based on LR

```

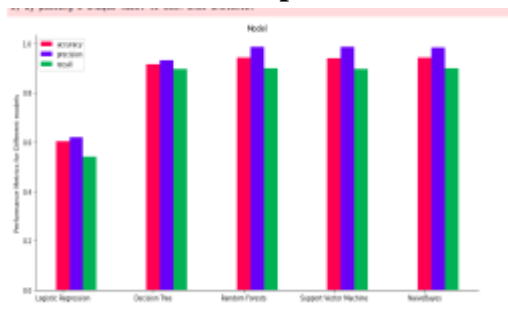
logit_pred = logit.predict(X_test)
pd.crosstab(pd.Series(y_test), pd.Series(logit_pred), name = 'Predict'), margins = True)

Predict  0  1  All
Actual
0  3465  1 1466
1  167  0 167
All 3632  1 1633

```

## Prediction based on DT

## Model Comparison



From the above the RF & SVM are giving better accuracy for prediction



#### 4. CONCLUSION

Our study focused upon the advancement and the correlation of various machine learning models that are used in order to look over hospital admissions dealing with the Emergency department. Each model that we looked into was generated using information gathered from various emergency departments. These 3 models were able to be constructed using 3 different techniques which were namely Naive Bayes, Random Forest Classifier, and Support Vector Machine. Out of the 3 models that we were able to analyze we found that the model which was generated using the SVM classifier was found to be more successful and accurate when compared to the other two models which were generated using Random Forest and Naïve Bayes. The 3 models that we had decided to look into all showed very similar and comparable results. We believe that these models can help many hospitals in facing the global problem of the overflow of patients in the Emergency Departments. They can also help us to increase the Patient Flow in hospitals and reduce crowding overall. We also believe that such models can be used in various other fields in the real world as well in order to monitor the

performance of various objects. There is so much we can use these models for in the real world and we believe that we can build upon these models for various use cases.

#### REFERENCES

- [1] Li JYZ, Yong TY, Bennett D Et al. Outcome of the interposition of an acute assessment unit in the general medical service of a tertiary teaching hospital. *Med. J. Aust.* 2010; 192:384–7.
- [2] Prakash, K.B. & Dorai Rangaswamy, M.A. 2016, "Content extraction studies using neural network and attribute generation", *Indian Journal of Science and Technology*, vol.9,no.22,pp.1-10.
- [3] O'Brien D, Williams A, Blondell K et al. Impact of streaming 'fast track' emergency department patients. *Aust. Health Rev.* 2006; 30: 525–32.
- [4] King DL, Ben-Tovim DI, Bassham J. Redesigning emergency department patient flows: application of Lean Thinking to health care. *Emerg. Med. Australas.* 2006; 18: 391–7.
- [5] Gardner RL, Sarkar U, Maselli JH et al. Factors associated with longer ED lengths of stay. *Am. J. Emerg. Med.* 2007; 25: 643–50.
- [6] Prakash, K.B. 2018, "Information extraction in current





Indian web documents",  
International Journal of Engineering  
and Technology(UAE), vol. 7, no. 2,  
pp. 68-71.

[7] Emergency Department  
Overcrowding in Massachusetts.  
Making Room in our Hospitals.  
Issue Brief.The Massachusetts  
Health Policy Forum, No 12; 2001.

[8] National Hospital Ambulatory  
Medical Care Survey. 2002  
Emergency Department Summary.  
Advance Data Number  
340.35pp.(PHS) 2004-1250.

[9] Prakash, K.B., Kumar, K.S. &  
Rao, S.U.M. 2017, "Content  
extraction issues in online web  
education", Proceedings of the 2016  
2nd International Conference on  
Applied and Theoretical Computing  
and Communication Technology,  
iCATccT 2016, pp. 680.

[10] K.S.S. Joseph Sastry & T.  
Gunashekar, ' A Systematic Access  
Through Machine Learning Methods  
For Expectation In Malady Related  
Qualities', International Journal of  
Engineering and Advanced  
Technology (IJEAT), Volume-8,  
Issue6S, August 2019, ISSN: 2249 –  
8958.