**ITCS 6122**

**SOFTWARE SYSTEM DESIGN AND IMPLEMENTATION**

**A PROJECT REPORT**

**ON**

**TITANIC: MACHINE LEARNING FROM DISASTER**

**TEAM MEMBERS**

Neeraj Kumar Nittori(800953520)

Nikita Kunchanwar(800962459)

Priya Kumari(800964015)

Shruti Bilgundi(800963974)

**TABLE OF CONTENT**

|  |  |
| --- | --- |
| **S.No** | **Topic** |
| 1 | Abstract |
| 2 | Introduction |
| 3 | Material and Methods |
| 4 | Software, Hardware and Technical Requirements |
| 5 | UML Diagrams |
| 6 | Sample Runs with Output |
| 7 | Result Analysis |
| 8 | Conclusion |
| 9 | Future Enhancements |
| 10 | References |

**ABSTRACT**

The task of this project is to predict whether a given passenger survived the sinking of the Titanic based on various attributes including age, gender, family size, ticket, the fare they paid, class and other information using tools of machine learning.

**INTRODUCTION**

Using data provided by www.kaggle.com, our goal is to apply machine-learning techniques to successfully predict which passengers survived the sinking of the Titanic. Features like ticket price, age, sex, and class will be used to make the predictions. Gender seems to be a strong indicator of survival, with women having a much better chance. It also takes class, fare, port, age, sex and family size into consideration to get an improved prediction.

Feature engineering (Data cleaning, Data filling etc) is applied on each of the above attributes to analyze the test and train dataset. Then train dataset form the input to Random forest model for training, which would predict the survival of passengers in test dataset.

**MATERIAL AND METHODS USED**

**Material Used:**

This project is based on data set provided by the problem on [https://www.kaggle.com/c/titanic](https://www.kaggle.com/c/titanic%20) and with our contribution, data-set was analyzed and useful insights to the problem were deduced.

**Data Set:**

The historical data has been split into two groups, a 'training set' and a 'test set'.

We were given 891 passenger samples for our training set and their associated labels of whether or not the passenger survived. For each passenger, we were given his/her passenger class, name, sex, age, number of siblings/spouses aboard, number of parents/children aboard, ticket number, fare, cabin embarked, and port of embarkation. For the test data, we had 418 samples in the same format. The dataset is not complete, meaning that for several samples, one or many of fields were not available and marked empty (especially in the latter fields – age, fare, cabin, and port). However, all sample points contained at least information about gender and passenger class. To normalize the data, we replace missing values with the mean of the remaining data set or other values.

1. PassengerId: Unique passenger identification number

2. Survived: Did a passenger survive or not (0 = died, 1 = survived)

3. Pclass: Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)

4. Name: Name of Passenger

5. Sex: Sex of Passenger

6. Age: Passenger Age

7. SibSp: Number of Siblings/Spouses Aboard

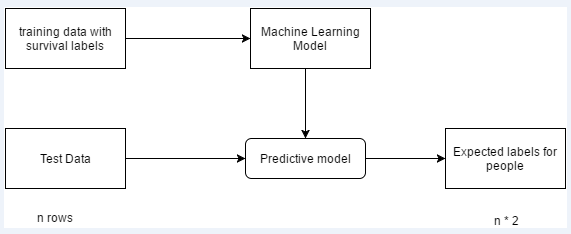
8. Parch: Number of Parents/Children Aboard

9. Ticket: Ticket Number

10. Fare: Passenger Fare

11. Cabin: Cabin

12. Embarked: Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

****

**Methods Used:**

**Modeling: Cross Industry Standard Process for Data Mining**

CRISP-DM breaks the process of [data mining](https://en.wikipedia.org/wiki/Data_mining) into six major phases.

The sequence of the phases is not strict and moving back and forth between different phases is always required. The arrows in the process diagram indicate the most important and frequent dependencies between phases. The outer circle in the diagram symbolizes the cyclic nature of data mining itself. A data mining process continues after a solution has been deployed. The lessons learned during the process can trigger new, often more focused business questions and subsequent data mining processes will benefit from the experiences of previous ones.

****

**Business Understanding**

This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition, and a preliminary plan designed to achieve the objectives. A decision model, especially one built using the Decision Model and Notation standard can be used.

**Data Understanding**

The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.

**Data Preparation**

The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include table, record, and attribute selection as well as transformation and cleaning of data for modeling tools.

**Modeling**

In this phase, various modeling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements on the form of data. Therefore, stepping back to the data preparation phase is often needed.

**Evaluation**

At this stage in the project, you have built a model (or models)that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to more thoroughly evaluate the model, and review the steps executed to construct the model, to be certain it properly achieves the business objectives. A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.

**Deployment**

Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that is useful to the customer. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data scoring (e.g. segment allocation) or data mining process. In many cases it will be the customer, not the data analyst, who will carry out the deployment steps. Even if the analyst deploys the model it is important for the customer to understand up front the actions which will need to be carried out in order to actually make use of the created models.

**Approach Used:**

**Random Forest-**

A random forest is a classifier consisting of a collection of tree-structured classifiers {h(x, Θk), k = 1.. k} where the {Θk} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x. A random forest is an ensemble of decision trees, which will output a prediction value, in this case survival. Each decision tree is constructed by using a random subset of the training data. After you have trained your forest, you can then pass each test row through it, in order to output a prediction. This particular python function requires floats for the input variables, so all strings need to be converted, and any missing data needs to be filled.

**SOFTWARE, HARDWARE AND TECHNICAL REQUIREMENTS**

**Software Requirements:**

1. Python 2.7

2. Anaconda 4.2.0

3. Packages Used: Pandas, NumPy, SciPy, Scikit-Learn, Matplotlib, Statsmodels

**Hardware Requirements:**

1. RAM: 8 GB

2. Processor: i3 or i5

3. 256 GB Hard Disk

**Technical Requirements:**

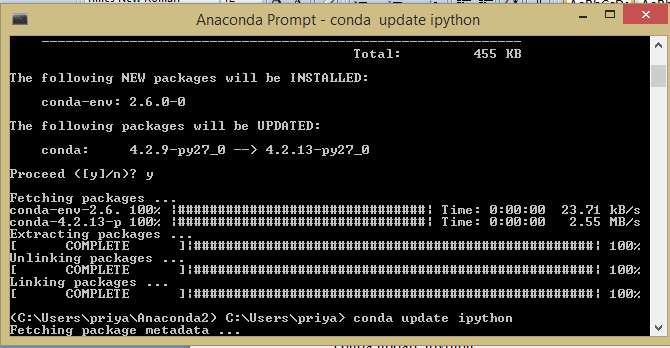
1. Download and install Continuum’s Anaconda or the free edition of Enthought’s Canopy.

Link: <https://www.continuum.io/downloads>

1. Update IPython to the current version by running the following commands in ‘Anaconda Prompt’:

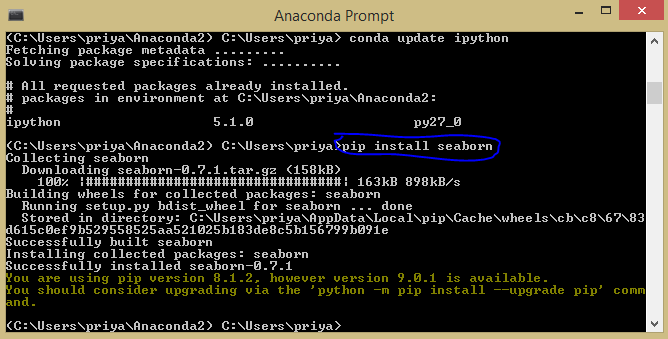
conda update conda

conda update ipython

****

1. Run the following command in Anaconda Prompt to install seaborn

Pip install seaborn



1. Launch Jupyter Notebook from windows.(The software gets installed with Anaconda Distribution).

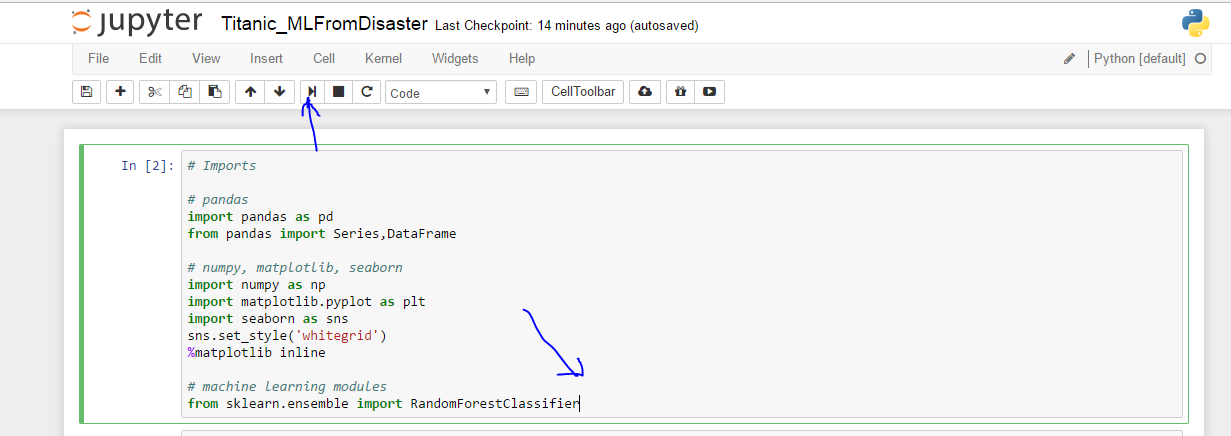
It opens a web browser where the file ‘Titanic\_MLFromDisaster.ipynb’ is to be placed along with two datasets: Train.csv and Test.csv



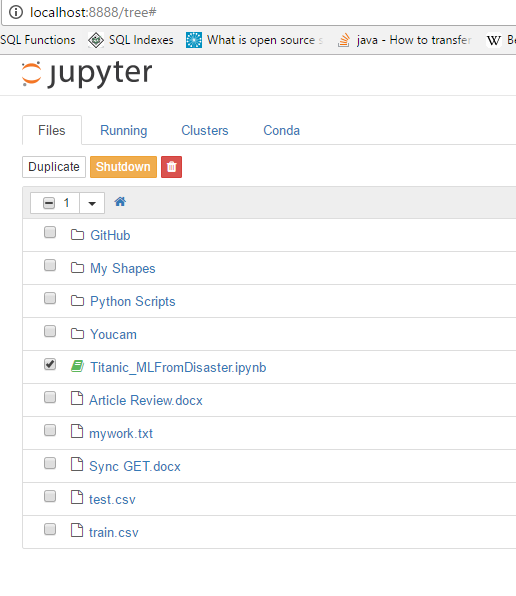


After placing the three files in the specified path of your system, double click on ‘Titanic\_MLFromDisaster.ipynb’ and run.

Place the cursor at the end of ln[2] and hit the run icon (as shown in the figure below).

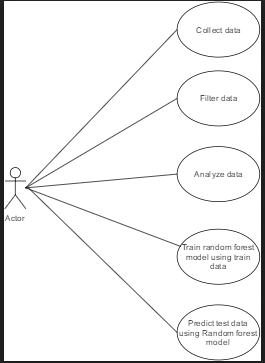


Keep running the program to get the analysis one by one til the output score is generated along with the output csv file ‘Titanic.csv’.

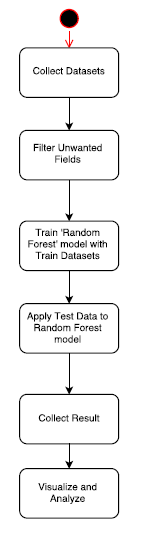


**UML DIAGRAMS:**

1. Use Case
2. Activity Diagram
3. Sequence Diagram
4. **Use Case Diagram**

****

1. **Activity Diagram**

****

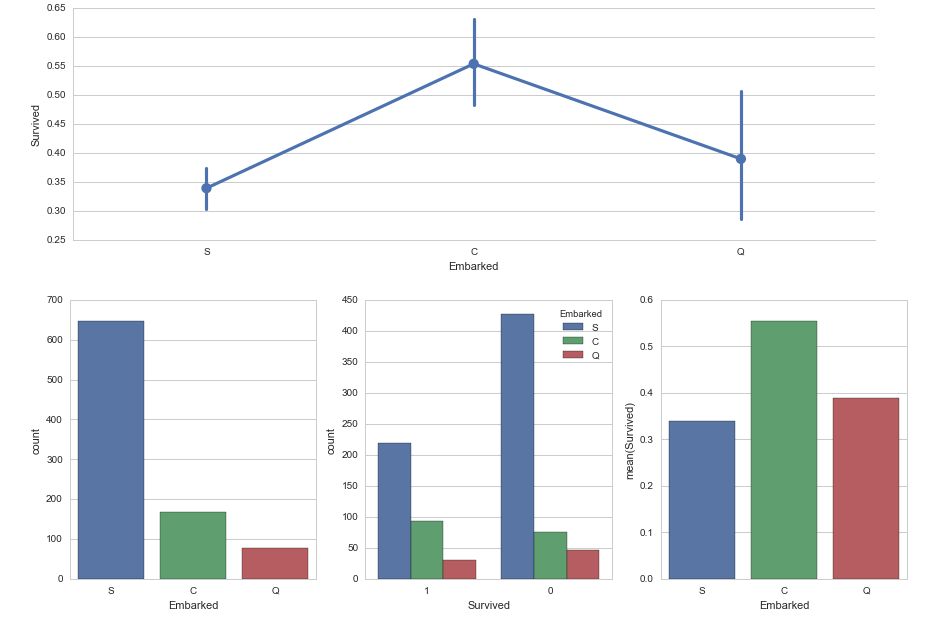
1. **Sequence Diagram**

**C:\Users\priya\Downloads\Sequence Diagram.png**

**Sample Runs with Output:**

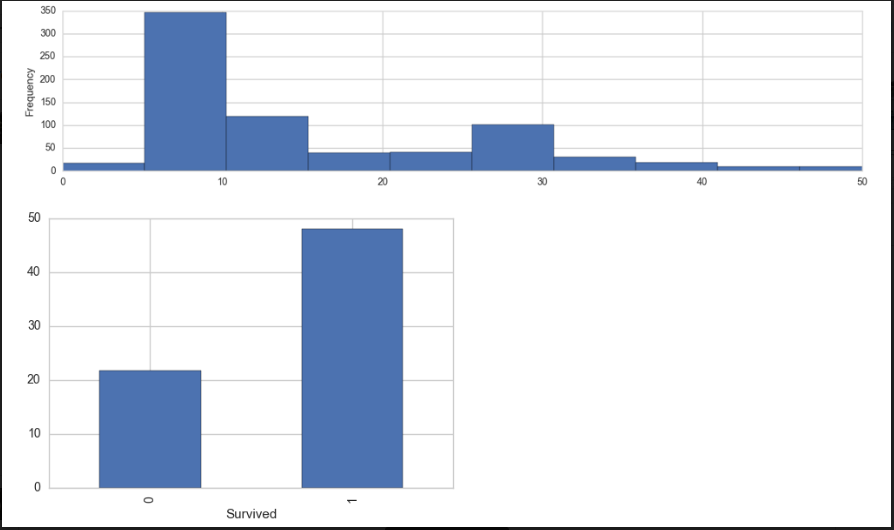
The feature extraction is kind of tricky, because each feature has different effects on the model and if combine some features together by multiply, square operation, they will have other effects. First of all, we observe every single feature.

Count of people belonging to S embark is .more than compared to count of people in C or Q. From the graph derived, we can say that rate of survival is lowest for S embark.

****

**Fig: Based on Embark**

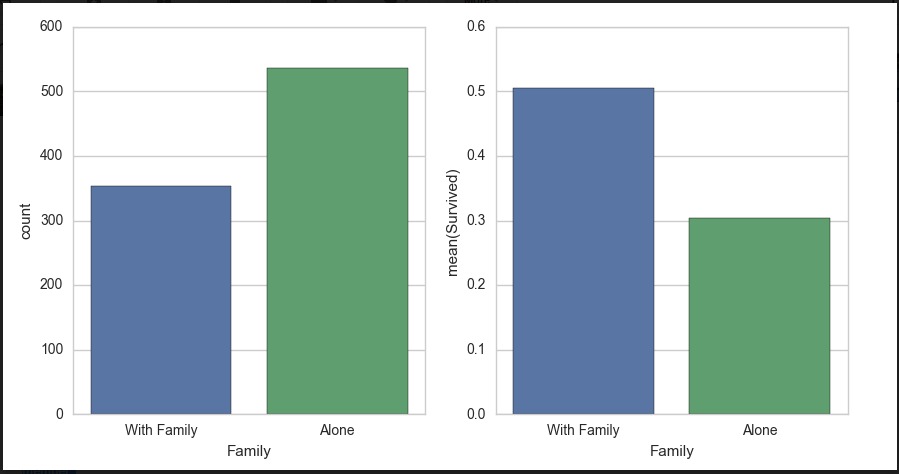
From our analysis, we found that the average **fare** of survived was 47.99 whereas the average fare of perished was 21.69. We can think about rich people get some privileges. So predicting rich people as survived is reasonable. But most of the fares lie between 0 to 50 dollars so we have to make do more to predict accurately.



**Fig: Based on Fare**

We have two columns ‘Parch’ and ‘SibSp’ in the input data showing if any parent (or child) and sibling exist respectively.  Instead of having two columns we can have only one column “Family” represent if the passenger had any family member aboard or not. This new column can be used to show whether having any family member will increase chances of survival or not.

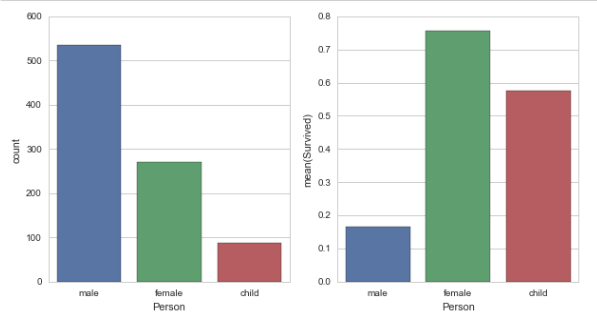
From below graph of ‘Family’ over ‘mean (survived)’ we can analyze that chances of survival increased when passenger had any family member.



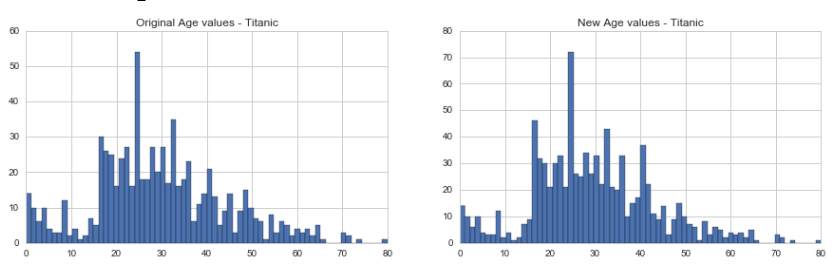
**Fig: Based on Family**

Obviously, gender is the most important element affecting survival rate.

We have derived ‘child’ person type for age <16 and then analyzed their survival rate along with male and female passengers. From the first graph shown below, we can say that count of males was higher than females and children. The second graph shows the relationship between mean of survived against person types. Male has lower survival rate than children and female. Female has the highest survival rate. So we can conclude that gender is the most important feature in predicting the survival rate.

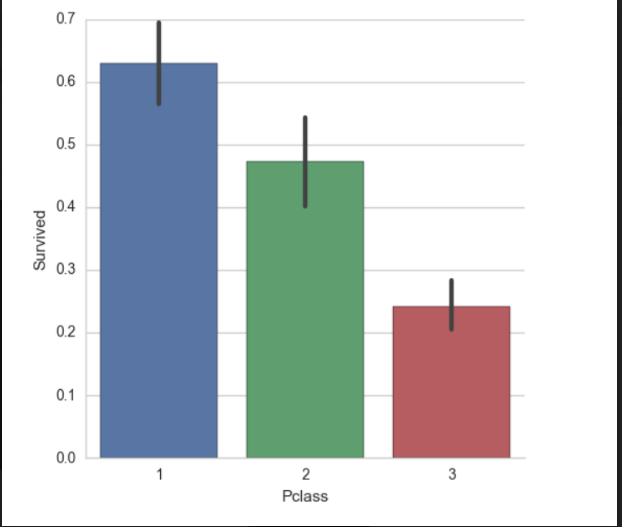
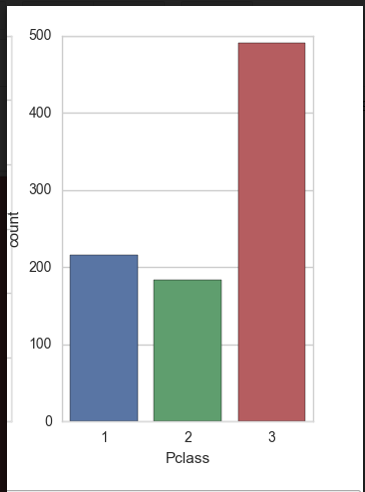


The histogram of 'Age' values in test and train data shows that the Age values are normally distributed. According to the Bell curve, 68% of data are within one standard deviation. Hence, for the records whose Age value is not available, we replace them with the random numbers within one standard deviation i.e., range (mean+standard deviation, mean-standard deviation). Finally, all the values are converted from float to int.

****

**Fig: Based On Age**

Pclass stands for passenger’s class. Count of passengers belonging to 3rd class is the highest but has the lowest survival rate**.**

****

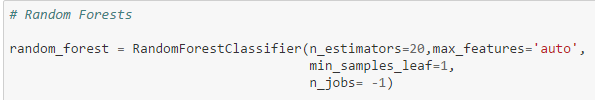
Cabin has a lot of NaN (not a number) values, so it won’t cause a remarkable impact on prediction.

**Random Forest Algorithm:**

We follow three simple steps: initializing the model, fitting it to the training data, and predicting new values.

A random forest is an ensemble of decision trees which will output a prediction value, in this case survival. Each decision tree is constructed by using a random subset of the training data. This random subset (Bootstrap sample) is typically drawn with replacement, meaning the same record can be drawn multiple times. After training the forest, we pass each test row through it, in order to output a prediction.

In our analysis, we have used modeled our Random Forest Classifier with the following parameters to produce accurate score.



1. **n\_estimator**

It indicates the number of decision trees in the forest. In our analysis, we have set it to 20.

1. **max\_features:**

It indicates the number of features to consider when looking for the best split.

1. **min\_samples\_leaf:**

The minimum number of samples at leaf node is 1.

1. **n\_jobs:**

The number of jobs to run in parallel for both fit and predict. If -1, then the number of jobs is set to the number of cores.

**CONCLUSION:**

In our project, random forest model correctly predicted the survival of 96% of the test data set. According to the output file generated, 163 passengers survived out of 418. We have successfully analyzed the data and predicted the survival.

**FUTURE ENHANCEMENT:**

1. In our project, we have used Random Forest algorithm as a tool of machine learning to predict survival rate. We can also use different tools of machine learning such as: Support Vector Machine, Gender Based Model to predict survival and compare the accuracy of results obtained.
2. A website or mobile application can be created to load and analyze data.
3. Different feature analysis can also be applied.
4. Insignificant variables can be identified for further model testing.

**REFERENCES**

<http://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html>

<https://www.kaggle.com/c/titanic/data>

<http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

<https://www.youtube.com/watch?v=OByOgGXq76A>

<http://scikit-learn.org/stable/modules/ensemble.html>

Useful Resources:

* Kaggle Tutorial and forums
* Feature Engineering