Machine Learning to Calculate Trip Budget

Megha Sharma

Abstract: *Trip planning requires effort. Majority of which is consumed in balancing preferences of travel and stay; with budget. This effort can be minimized using budget estimator. Summing up the total costs to calculate budget is ideally correct. Practically, budget can differ from individual to individual based on their nature. Some prefer to spend more while some less. Machine Learning could help predict human nature using feedback mechanism. Taking feedback about total cost incurred and comparing it to actual estimate could give insight about user nature to the system. In this paper, we have built a budget estimator that considers user preferences and uses regression algorithm to compute costs. It later asks user to input the actual cost incurred, correcting its previous estimate and uses the updated entry to drive data to be more user-specific. The system gives percent classification of 84% and percent recognition of 72.27%.*

Keywords: *Budget Estimator, Feedback, Machine Learning, Percent Classification, Percent Recognition*

I. INTRODUCTION

People often tend to travel to different places to enjoy scenic beauty or to explore the adventure, the place offers. Most people tend to travel at least once in a year. They escape from their normal routine to pursue something that could give them immense pleasure. The reason could be to relax i.e. to reduce stress, or to make memories, or for fun, or to see more of world. For average people, trip is not a one-night process. They start months in advance selecting their destination and then start saving portion of their earnings for upcoming trip. One needs to have a realistic estimate else, one might risk living the entire experience being worried about the money spent. If not, one might spend whole-heartedly and end up spending more than they expected. This might lead to stress and leave bad memories as an aftermath - the one thing person tries to escape by taking a trip. Thus, it kills the sole purpose of travelling in first place. People often turn to travel agencies to escape these scenarios. Travel agencies are profit-making institutions. With little effort and proper planning, the money spent in travel agencies can be invested in some other want. The latter also allows you to finalize itinerary as per your choice. Additionally travel agencies know little about you and so can only speak for ideal costs incurred and not the total. Their offers are based on these ideal costs. An insight into personal nature is required, to help calculate proper estimate. Adding on, while planning it

often occurs, one starts with 5-star hotels as their preferred choice of stay, then search for those hotels. If they find the place expensive, they revise their preference to 4-star and possibly continue the same to 3-star and then to 2-star. A budget calculator can help minimize this effort thus making the planning easier. With a few clicks, you can get an estimate whether your choice is affordable or not. Thus, saving hours of time spent on search engines.

In all, a ML based budget calculator can provide below benefits:

1) Provide you with realistic total cost estimate based on your preferences and persona
2) Estimates for you, monthly savings needed for upcoming trip
3) In cases where your savings fail to meet estimated goal, it saves your time, by giving revised cost in minutes (based on your revised preferences)
4) It sets itself according to your past travel experiences and provides with a more suitable estimate next time

II. LITERATURE SURVEY

Origins of estimation can be traced back to the days when trade was initiated. It is through this art of planning that humans are able to save and produce their dreams to reality. Some have developed the talent of saving while still living their dreams. As for others, they still struggle in understanding how much to spend. For a successful travel journey, an estimate about total cost is required. It can be calculated by humans or by a machine. It is the end product - a correct estimate, that matters. A travel budget helps one plan better and gives way to stress-free holiday. It helps you create a realistic travel plan, acts as a guide to set saving goals and helps make the best spending choices on the run[1].

III. DATASET

The dataset used is created by running surveys on 61 different people noting about their most recent trip experience. Data being the most crucial aspect of ML, its correctness can make or break an application [2]. Training data needs to be latest in order to get correct estimates. This is because currency values tend to differ every moment, resulting in upsurge or decline of hotel bookings, travel tickets and restaurant costs. Consequently, affecting the overall estimate. As 61 records were collected, they are further split into training set containing 50 records and testing set of 11 records. Fig 1. Depicts the analysis on data collected through survey.
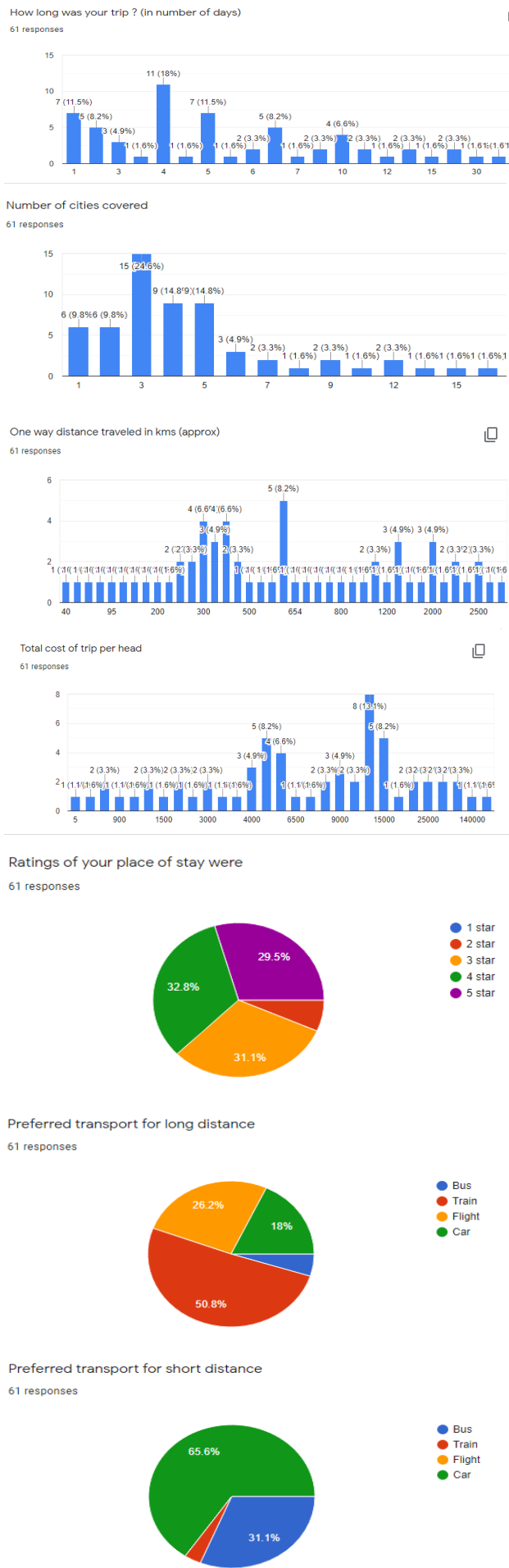
Fig. 1.Analysis of data collected from survey

IV. METHODOLOGY

Gathered data is processed to convert text data into number format. Fig 2. enlists all steps that need to be performed on data. As the system will include a factor for natural tendency of user through feedback, an additional feature is added to account for additional theta in gradient descent. The objective of linear regression is to minimize cost function J, where J can be defined as :

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2 \quad (1)$$

Where,

J represents cost function
m represents number of records
x^i represents i^{th} feature record in feature set
$h_\Theta(x)$ represents the estimate function of x
y^i represents actual trip budget of i^{th}record in features

In our problem, $h_\Theta(x)$ can be formulated as :

$$h_\theta(x^i) = \theta_0 + \sum_{j=1}^{n}(\theta_j * x_j^i) \quad (2)$$

Where,

$h_\Theta(x^i)$ represents the estimate function of ith record
n represents number of features
x_j^i represent j^{th} feature of ith record
Θ values represents paramters used to fit the data

Eq. 2 can be simplified by adding additional feature as :

$$h_\theta(x^i) = \sum_{j=0}^{n}(\theta_j * x_j^i) \quad (3)$$

Where,

$x_0^i=1$, for all x^i

Eq. 3 can be re-written as :

$$h_\theta(x^i) = \theta^T X \quad (4)$$

Where,

Θ^T represents Θ transpose
X represents (m x n) matrix of x

Gradient descent helps fit the parameters according to dataset. Its formulas can be represented as below :

$$\theta_j = \theta_j - \frac{\alpha}{m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)x_j^i \quad (5)$$

Where,

m represents number of records
j represents the index in feature set
α represents learning rate
x_j^i represent j^{th} feature of ith record
$h_\Theta(x^i)$ represents the estimate function of ith record
y^i represents actual trip budget of i^{th}record in features

Thus, for our problem, we have theta dimension of (7x1)

3	3 star	Car	Car	2	1200
10	3 star	Train	Car	5	2500
6	5 star	Train	Car	3	750
7	3 star	Train	Bus	4	1500
15	5 star	Flight	Car	3	600
7	4 star	Train	Car	3	987

Output Y

Features X

3	3 star	Car	Car	2		1200
10	3 star	Train	Car	5		2500
6	5 star	Train	Car	3		750
7	3 star	Train	Bus	4		1500
15	5 star	Flight	Car	3		600
7	4 star	Train	Car	3		987

Format to machine readable form

Feature scaling and mean normalisation

3	3	4	4	2		1200
10	3	2	4	5		2500
6	5	2	4	3		750
7	3	2	1	4		1500
15	5	3	4	3		600
7	4	2	4	3		987

Feature scaling and mean normalisation

-0.168	-0.291	0.555	0.338	-0.116
0.199	-0.291	-0.111	0.338	0.008
-0.010	0.375	-0.111	0.338	-0.074
0.041	-0.291	-0.111	-0.661	-0.033
0.462	0.375	-0.111	0.338	-0.074
0.041	0.042	-0.111	0.338	-0.074

Add in feature0 for theta0

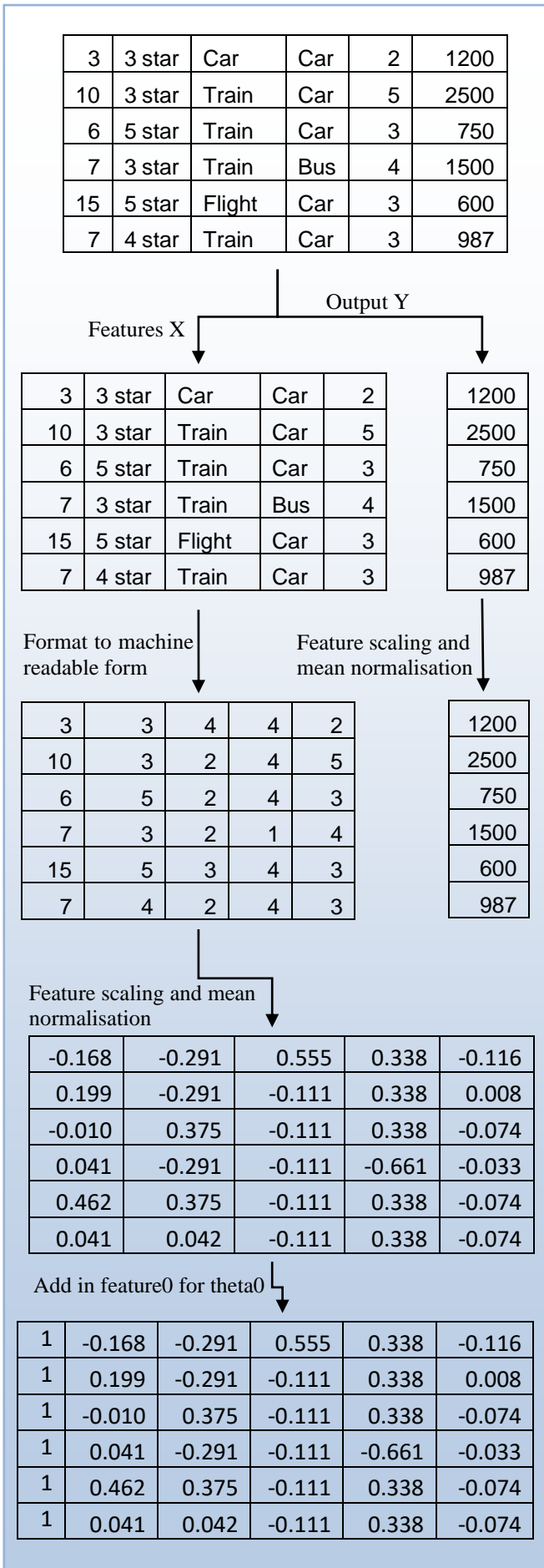| 1	-0.168	-0.291	0.555	0.338	-0.116
1	0.199	-0.291	-0.111	0.338	0.008
1	-0.010	0.375	-0.111	0.338	-0.074
1	0.041	-0.291	-0.111	-0.661	-0.033
1	0.462	0.375	-0.111	0.338	-0.074
1	0.041	0.042	-0.111	0.338	-0.074

Fig. 2.Block diagram representing pre-processing steps

During training, we let gradient descent compare features and actual cost. Based on these inputs it calculates theta values. These theta values can then be used to predict estimated cost from the user preferences.

The proposed methodology can be formulated as :

1. Pre-processing of data
   1.1 Separate data into feature set X and output set Y
   1.2 Convert data into machine-readable format - all numbers
   1.3 Calculate new values for X and Y using feature scaling and normalization
   1.4 Add in additional feature for accounting for human factor
2. Processing of data
   2.1 Compute gradient descent for various combinations of learning rate and number of iterations
   2.2 Finalize the parameters based on minimum value of cost function
   2.3 Use these values of alpha and iteration for gradient descent to fit feature and output set; in order to obtain theta values
3. Compute results
   3.1 Follow pre-processing steps for testing set
   3.2 With the obtained theta values calculate the actual total estimate

V. EXPERIMENTAL RESULTS

To verify the correctness of idea, each feature was plotted independently on x-axis and total cost on y-axis. From Fig. 3 it is evident that, for most of the records the max price limit can be predicted linearly.

As features are collected from different individuals the initial phase would have a minimal error, i.e. similar values of features can result in different values of estimate. This error would reduce as system adapts to single user.
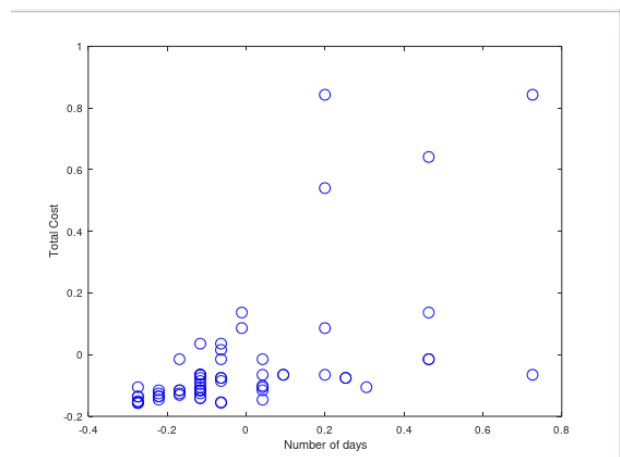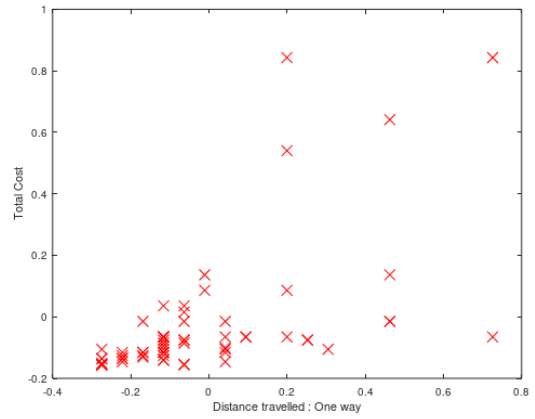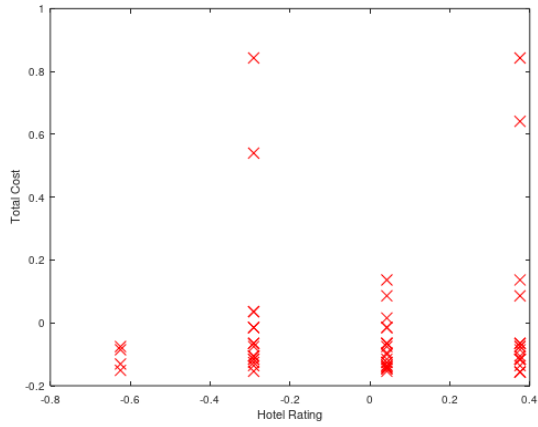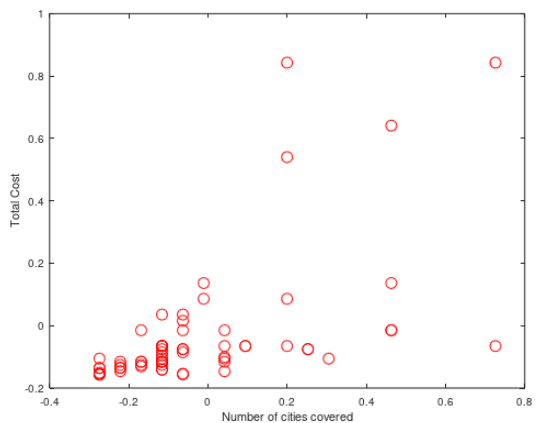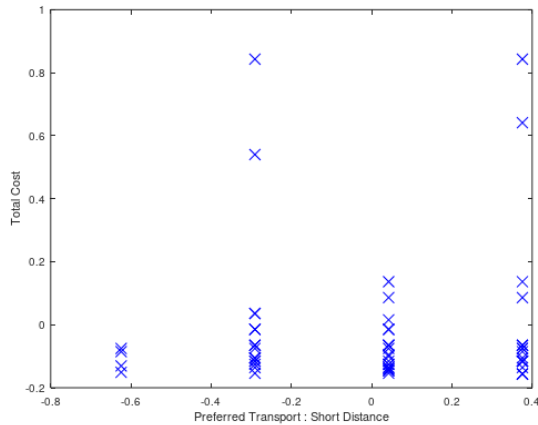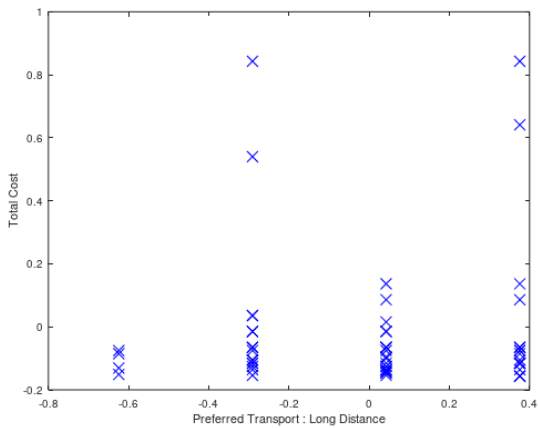
Fig. 3.Depicts individual features on x-axis apped with total estimate on y-axis

Deciding learning rate and number of iterations is crucial in gradient descent. We have to select values such that gradient descent reaches theta values that have minimal value of cost function. Gradient descent should be able to reach a conclusion efficiently i.e. requiring less number of steps, at the same time learning rate should not be big enough, else it might fail to converge. Table I depicts valrious values of learning rate and iterations for our dataset.

Table- I: Gradient Descent Results

Learning rate	No. of iterations	Min. value of J
0.01	500	0.03
0.03	500	0.029
0.05	500	0.020
0.06	500	0.018
0.1	500	0.015
0.5	500	0.012929
0.9	500	0.012903
1.3	500	0.012902
0.9	100	0.0139
	200	0.0130
	300	0.01319
	400	0.012904
	500	0.012903
1.3	100	0.013313
	200	0.012923
	300	0.012903
	400	0.012902
	500	0.012902
1.5	200	0.012910
	300	0.012903
1.7	200	0.012905
	300	0.012902
1.9	200	0.12904
	300	0.12902

Based on the above results it was observed, the most effective results are given when either of the below parameters were selected:
1) Learning rate : 0.9, Number of iterations : 400
2) Learning rate : 1.9, Number of iterations : 200

Using the above values of variables, gradient descent was implemented to calculate value of theta. Further, using Eq.4 budget estimate was calculated.

Table- II: Application Results

Min. Value of cost function	Observed theta	Percent classification	Percent recognition
0.012904	[-0.021198, 0.576035, 0.044818, -0.051915, 0.114076, -0.179519, 0.010560]	84.00	72.27

VI. ALGORITHM

Application algorithm can be formulated as :

1. If first time user : save user details - name, email-id and password
   Else : Ask for login details
2. If user has no previous trips : skip this step
   Else if user has given feedback for last trip : skip this step
   Else : As a part of feedback, ask user to enter the actual total cost of trip.
      If feedback is provided :
         Update the entry in stored estimate set;
         Merge the updated estimate set entry to training set;
3. Train the machine using gradient descent on training set
4. Ask user for input, consider this as testing data
5. Calculate results and store in estimate set.
6. Ask for user input, on planned date
7. Calculate a monthly saving estimate using current date and planned date

VII. FUTURE SCOPE

Budget estimator currently considers six features but can be expanded to include:

1) Number of metro-cities
   a. It will help give a better cost estimate as prices in metro-cities are usually higher than other cities/villages
2) List of cities travelled
   a. This feature could replace the feature 'distance travelled: one way', as it gives accurate distance travelled instead of an estimate.
   b. Distance between two cities would be calculated using web scraping
   c. Overall distance could be calculated by summing up distances of two consecutive cities in the list
3) List of adventure sports or fun activities undertaken
   a. These activities cost more and so could provide with better understanding about estimate
4) Currency unit and change factor
   a. It can aid in expanding scope of budget estimator

VIII. CONCLUSION

Budget estimator promises a packaged application flow. It is easy and simple for both - use and implementation. Its application can aid humans in planning their trip by estimating costs according to their preferences. The system is unique because it considers user's nature as factor in predicting costs. With help of this system, users can compare estimates by changing preferences and running the algorithm again. It would give them a better understanding about the aspect of trip that could be compromised for saving costs. It can also aid users by giving an insight about monthly savings needed for the travel, based on estimates provided.

The findings of initial run i.e. percent classification of 84% and recognition of 72.27% reflect the goodness of the system. Its accuracy can be enhanced by including more features like number of metro-cities, list of cities travelled, list of adventures or fun activities covered, etc.. Additionally, system's ability to adapt to a specific user promises to produce better results over time.

REFERENCES

1. Title: How to create a realistic travel budget that actually works. Available : https://budgetbakers.com/blog/create-realistic-travel-budget#:~:text=If%20you're%20still%20in,spending%20choices%20on%20the%20go.
2. Title: Preparing your dataset for machine learning : 8 basic techniques that make your data better. Available : *https://www.altexsoft.com/blog/datascience/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/*

AUTHORS PROFILE

Megha Sharma. She completed her B.Tech degree in Information Technology from Vishwakarma Institute of Technology, Pune in year 2019. She has studied at University of Western Ontario for duration of a semester. Her inclination towards computer science was not limited to completion of a graduate degree. Even working as a graduate analyst at Barclays Global Service Centre India, she continued to pursue her interests in research areas like Image Processing, Machine Learning and Human Computer Interaction. Her previous research work constitute of papers titled 'Iris Biometric Security and Fisher Based FER for Smart Homes' and 'Uber App Design : A Case Study of Human Computer Interaction Enhancement'.

Retrieval Number: J73890891020/2020©BEIESP
DOI: 10.35940/ijitee.J7389.0891020
Journal Website: www.ijitee.org

54

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication