World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Predicting waiters prompt service by analyzing restaurants rating and other factors using machine learning

Sunil Bhutada, Chandana Kavuri *, Sanjana Marru and Anusha Tanniru

Department of IT, Sreenidhi Institute of Science & Technology (a) Hyderabad, Telangana, India.

World Journal of Advanced Research and Reviews, 2022, 15(01), 064–074

Publication history: Received on 10 May 2022; revised on 12 June 2022; accepted on 14 June 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.15.1.0552

Abstract

This project aims to predict waiters prompt service and analyze how other factors like Restaurant's ranking, Ambience, customers finance etc., effect waiters small incentive. Major reasons were analyzed so, that restaurant brings some changes to increase waiter's service along with restaurant's reputation and it's like giving good reward for their appreciative service. Not only waiters service effect their incentive but also restaurants ambience, ranking, food quality also because some effect. Some models were proposed to predict the tip these shows the result with all the factors involved and helps to predict the expected result. The proposed model is validated against techniques like Random forest Regressor Using Hyper tuning, Bayesian Ridge Regressor, Elasticnet Regressor. Along with good visualization for better analysis using Mat plot, seaborn. They are particularly suited to predicting exact output as expected. For implementation purposes, choose features like total_bill, tip, sex, smoker, day, time, etc., the proposed model is evaluated with a waiter's tip data set along with some changes to dataset based on various performance to show its effectiveness.

Keywords: Prompt Service; Regressor; Ambience; Customer Finance; Food Quality; Techniques; Analysis

1. Introduction

The pace with which waiters deliver meals depends on a number of circumstances, including the fine dining restaurant type, the number of persons in your parties, the money you paid on the bill, the food quality, and the ambiance of the restaurant. Waiter Tips Analysis is a prominent data science case study in which we need to forecast the tips offered to a waiter in a restaurant for providing food. The ranking of a restaurant is determined by a variety of factors. For example, if the restaurant's food quality and service are excellent, people are more likely to visit it, and the restaurant's score will rise as a result. If a restaurant has the highest rating, waiters will provide faster service. We can figure out which restaurant gets the highest rating based on many factors using this forecast. Prompt service is a practical method to express your gratitude for your server's efforts to make your dinner enjoyable, often known as waiter's timely service. A restaurant's ambiance is equally vital. A lot of factors influence waiters' prompt service, along with the quality of the meal, the ambience, and the hotel's ranking. People frequently check restaurant ratings and reviews to determine which restaurants have the best ratings and give excellent service. If the rating is high, it indicates that the restaurant's size will expand. As a result, knowing which restaurant has the greatest ranking based on which characteristics are critical. Waiters' timely service is also determined by the restaurant's ranking. As a result, we're using data modeling to predict tips based on the environment and to improve tips by making some changes.

In the restaurant business in the United States, the tipping structure has evolved significantly. There were no predefined guidelines on how well the tip amounts should be determined when tipping was first introduced to the United States, and gratuities were designed to be a method for clients to express thanks to waiters for their exceptional services.

* Corresponding author: Chandana Kavuri
Department of IT, Sreenidhi Institute of Science & Technology(a) Hyderabad, Telangana, India.

However the tipping industry has evolved and grown to a norm significant enough to affect government taxes, as servers now rely on tips to Customer satisfaction (TSAT) Evaluation of Service Quality (SQ) Evaluation of Product Quality (PQ) Price 24 supplement their earnings (Lynn, 2006). In the United States, what began as a manner of expressing gratitude and respect to the waiter has evolved into a way of life.

In most circumstances, the traditional notion of customer happiness and tipping may be significantly less applicable now that tipping is the norm and expected regardless of other considerations .Tips in the restaurant sector, like incentives in other industries, are both an incentive and, to some extent, anticipated; however, tipping is done between the client and the server, but not between server and the manager. When other confounding factors are adjusted for, tips are a means to recognize good service.

There is no way of knowing how the tipping business will develop next, since it has evolved from a simple practise of paying servers for providing quality services that clients were pleased with to the complex activity it has become. Tipping has been examined in finance, sociology, ethics, and a variety of other fields. Tipping is not really a single phenomenon, noted Azar, O. (2004).

The social phenomena of tipping have been thoroughly explored now that it is a tangible standard. Many of these research have already been discussed. While scholars have looked into tipping from a variety of perspectives, including history, motivation, and more, most studies have focused solely on tipping in the framework of the server-customer interaction. This interaction, however, is simply one aspect of a bigger picture of what drives the restaurant business, including enabling customer satisfaction and generating revenue.

2. Methodology

2.1. Import Libraries

2.1.1. NUMPY

NumPy is a library that contains multidimensional array gadgets as well as a set of procedures for processing them. NumPy can be used to conduct arithmetic and scientific operations on arrays.

2.1.2. PANDAS

A Python facts analysis package. Pandas is based on the most advanced middle Python modules, including matplotlib for data visualization and Numpy for arithmetic operations. Pandas acts as a wrapper around these libraries, allowing you to access many of matplotlib's and NumPy's approaches with significantly less code.

2.1.3. SEABORN

A Python library piled on top of matplotlib library that is freely available. For data visualization and exploratory data analysis, it's widely utilized. Data frames and the Pandas libraries are no problem for Seaborn. The graphs that are created can also be customized without difficulty.

2.1.4. MATPLOTLIB

Matplotlib is a graphical library for Python and its NumPy numerical arithmetic extension. It provides an object-oriented API for embedding visualizations in packages using popular GUI tool kits like as Tkinter, wxPython, Qt, or GTK.

2.1.5. SCIP

SciPy is a systematic Python library that is open source and BSD-licensed for mathematics, science, and engineering.

2.1.6. SKLEARN

Sklearn is the most usable and robust Python package for system learning. Through a Python consistency interface, it disseminates effective tools for device learning and statistical modelling, such as type, regression, clustering, and dimensionality reduction.

2.1.7. MATH

The math package is a standard Python module that is always available. To use the mathematical functionality of this module, you must first import it using import math. It provides access to the C library's fundamental functionality. # Rectangle root computation import math math.sqrt, for example (4).

2.2. Importing files and libraries

```
[ ] import numpy as np
    import pandas as pd
    import matplotlib.pyplot as plt
    import seaborn as sns
    import io
    import math
    import sys
    import warnings
    wt=pd.read_csv(io.BytesIO(uploaded['tips1.csv']))
```

Figure 1 Importing Libraries

Here, we imported files like our dataset and libraries

Here we imported numpy, pandas, seaborn and matplotlib.

We imported libraries such as numpy as np, pandas as pd and seaborn as sns

As well, we imported and read the dataset which is in the CSV form using pd.read_csv. We know our dataset is in the form of CSV which is known as comma separated values.

2.3. Reading the dataset

```
wt
```

	total_bill	tip	sex	smoker	day	time	size	restaurant	Rank	Serving	waiters patience level	customer finance
0	16.99	1.01	Female	No	Sun	Dinner	2	a	1	avg	4	medium
1	10.34	1.66	Male	No	Sun	Dinner	3	b	4	good	2	medium
2	21.01	3.50	Male	No	Sun	Dinner	3	a	1	excellent	1	high
3	23.68	3.31	Male	No	Sun	Dinner	2	a	1	good	2	medium
4	24.59	3.61	Female	No	Sun	Dinner	4	d	3	bad	5	low
...	...	...	...	...	...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3	d	3	good	3	high
240	27.18	2.00	Female	Yes	Sat	Dinner	2	d	3	bad	5	low
241	22.67	2.00	Male	Yes	Sat	Dinner	2	d	3	bad	5	low
242	17.82	1.75	Male	No	Sat	Dinner	2	d	3	bad	5	low
243	18.78	3.00	Female	No	Thur	Dinner	2	d	3	good	3	high

244 rows × 12 columns

Figure 2 Dataset

The above dataset consists of 244 rows and 12columns

Here, after reading the dataset as we have 4 types of restaurants which we named as ‘a’, ‘b’ , ‘c’, ‘d’.

Firstly, we took the restaurant ‘a’ and we compared all the features with that restaurant.

```
grouped = wt.groupby(wt.restaurant)
wt_new = grouped.get_group("a")
wt_new
```

	total_bill	tip	sex	smoker	day	time	size	restaurant	Rank	Serving	waiters patience level	customer finance
0	16.99	1.01	Female	No	Sun	Dinner	2	a	1	avg	4	medium
2	21.01	3.50	Male	No	Sun	Dinner	3	a	1	excellent	1	high
3	23.68	3.31	Male	No	Sun	Dinner	2	a	1	good	2	medium
5	25.29	4.71	Male	No	Sun	Dinner	4	a	1	good	3	high
10	10.27	1.71	Male	No	Sun	Dinner	2	a	1	avg	4	medium
...	...	...	...	...	...	...	...	...	...	...	...	...
215	12.90	1.10	Female	Yes	Sat	Dinner	2	a	1	avg	3	medium
216	28.15	3.00	Male	Yes	Sat	Dinner	5	a	1	good	2	high
217	11.59	1.50	Male	Yes	Sat	Dinner	2	a	1	bad	5	low
218	7.74	1.44	Male	Yes	Sat	Dinner	2	a	1	avg	4	medium
219	30.14	3.09	Female	Yes	Sat	Dinner	4	a	1	avg	4	medium

63 rows × 12 columns

Figure 3 Gathering according to restaurant 'a'

```
[ ] wt_new.sum(axis=0,skipna=True)

total_bill                                                    1334.4
tip                                                           202.05
sex                 FemaleMaleMaleMaleMaleFemaleMaleFemaleFemaleMa...
smoker              NoNoNoNoNoNoNoNoNoNoNoNoNoNoNoNoYesNoNoYesNo...
day                 SunSunSunSunSunSunSunSatSatSatSatSatSatSatSunS...
time                DinnerDinnerDinnerDinnerDinnerDinnerDinnerDinn...
size                                                             166
restaurant          aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa...
Rank                                                              63
Serving             avgexcellentgoodgoodavgavgbadbadavggoodbadexce...
waiters patience level                                           187
```

Figure 4 Sum of values of restaurant 'a'

By using swarm plot we represent our plot using features restaurant on y-axis and Rank on y-axis.

We observe restaurants with their ranks by observing we got to know that Restaurant 'a' has highest ranking-1, restaurant 'b' has ranking-4, restaurant 'c' has ranking -2, restaurant 'd' has ranking –3.

2.4. Data Cleaning

Data that we collect contains many duplicate data such as null values, missing data, incorrect data and outliers. By containing all of these in the data we cannot train our data so that we need to clean our data. Data cleaning is a process of removing duplicate and incorrect data within a dataset. While the data is collected the data need to be analyzed properly. So, data cleaning is a process where our data gets prepared for data analyze is by removing any unwanted, incorrect and duplicate data. It is the process where we erase the information to make space for new data but find dataset accuracy without deleting information.

```
[ ]  wt.shape

     (244, 12)

[ ]  wt.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 244 entries, 0 to 243
     Data columns (total 12 columns):
      #   Column                  Non-Null Count  Dtype
     ---  ------                  --------------  -----
      0   total_bill              244 non-null    float64
      1   tip                     244 non-null    float64
      2   sex                     244 non-null    object
      3   smoker                  244 non-null    object
      4   day                     244 non-null    object
      5   time                    244 non-null    object
      6   size                    244 non-null    int64
      7   restaurant              244 non-null    object
      8   Rank                    244 non-null    int64
      9   Serving                 244 non-null    object
      10  waiters patience level  244 non-null    int64
      11  customer finance        244 non-null    object
     dtypes: float64(2), int64(3), object(7)
     memory usage: 23.0+ KB
```

Figure 5 Dataset shape and information

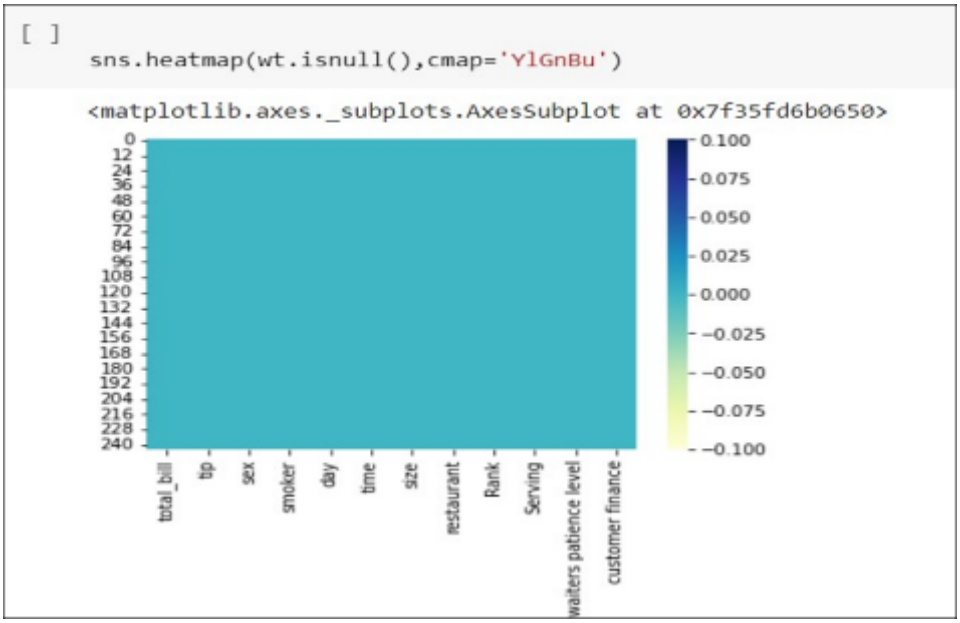

Figure 6 Null values

By using this heat map we can know that which features has null values.

As the colour of the heat map is constant there are no white spaces we can say that our dataset doesn't contain any null values.

2.5. Converting Categorical Type to Numerical

The dataset which we use for implementation contains both numerical as well as categorical data. Numerical data is in the form of numbers like 0,1 etc. Whereas categorical data are of three types which are classified as ordinal, nominal, and Boolean. This categorical data is in the form strings where we humans may understand but in the case of machines, they cannot interpret this categorical data directly. So we need to convert our categorical type data into numerical therefore, machines can understand. So, here we replaced our features from categorical to numerical values Feature

day: if it is sat as 0, sun as 1, thurs as 2, fri as 3.

Serving: if it is bad as 0, good as 1, avg as 2 and excellent as 3.

Time: if it is dinner as 0, lunch as 1.and so on

2.6. Data Visualization

Data visualization is a step in the data science process that must be completed once the data has been collected, processed, and modeled. To draw conclusions, the information must be visualized. The depiction of data in a graphical form is known as data visualization. Data visualization tools make it easier to observe and comprehend trend, outliers, and data patterns by including visual components such as charts, graphs, and maps.

Here are some examples of visualization tools and elements:

Tables, histograms, heat maps, plots, swarm plots, graphs, bar plots, bar graphs etc.,

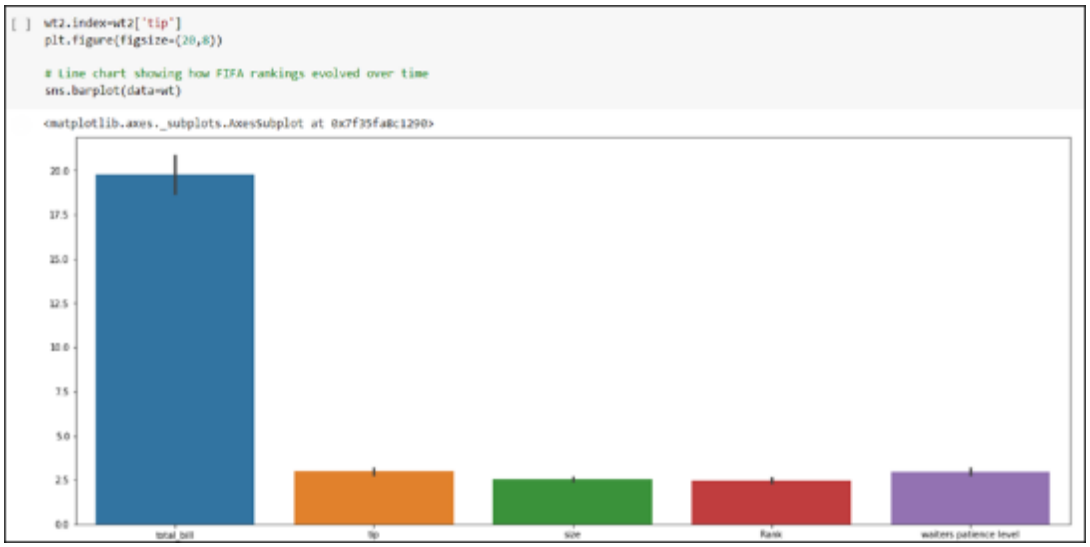

Figure 7 Comparing factors using colour differentiation

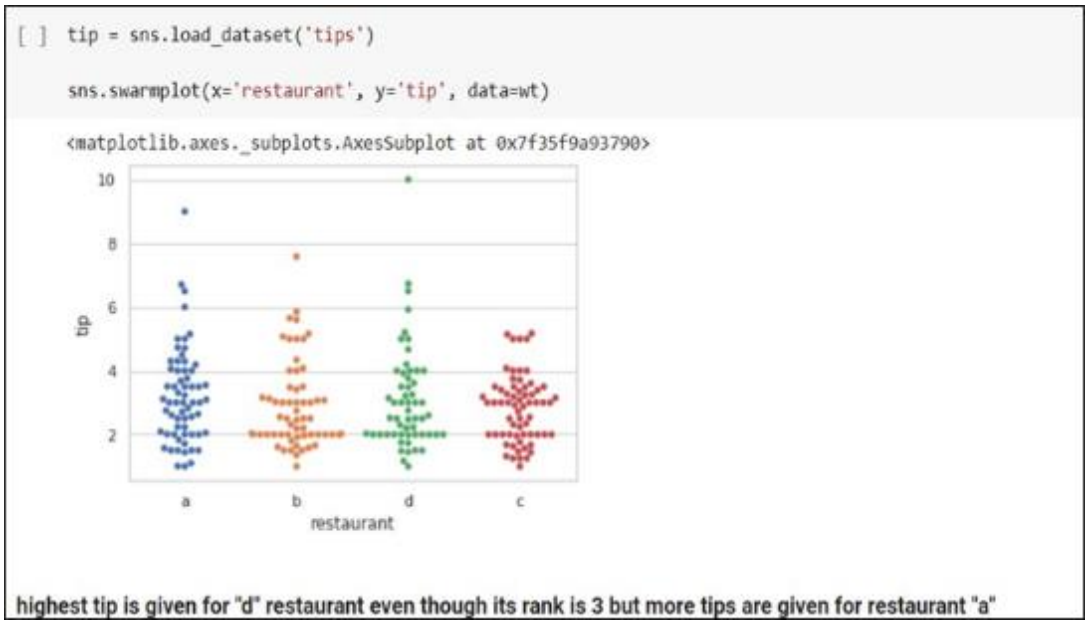

Figure 8 Restaurant vs tip

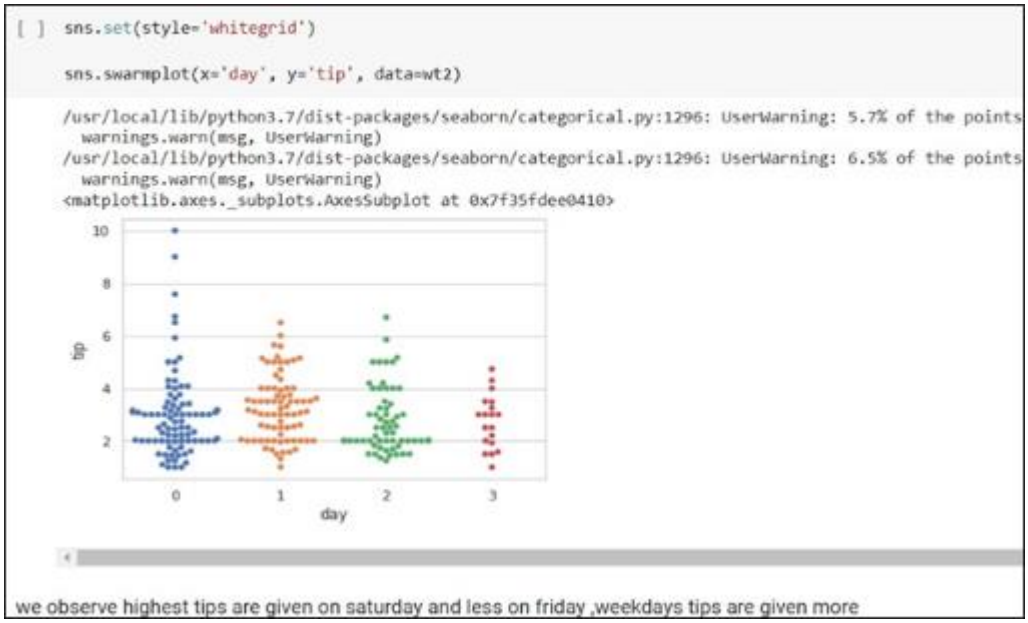

Figure 9 day vs tip

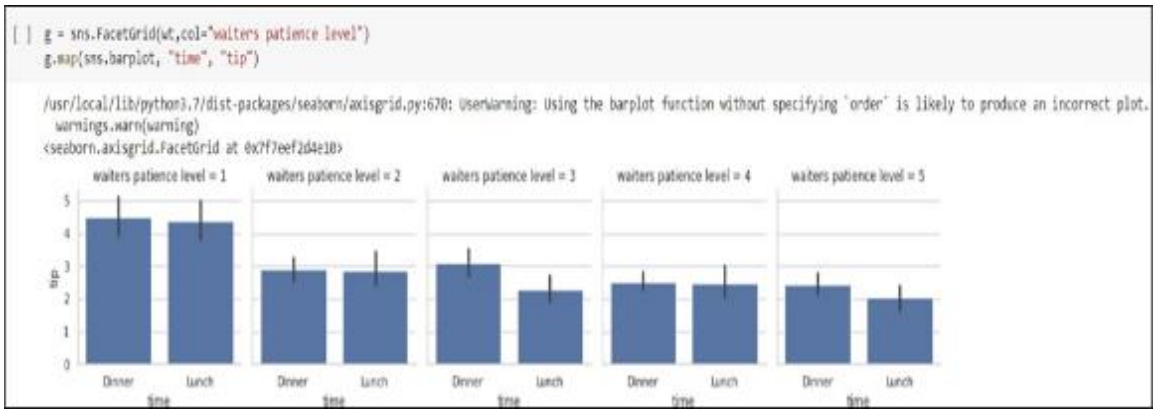

Figure 10 Comparing waiters patience level overtime vs tip

If patience level is more then waiter gets more tips and they get more tips during dinner

2.6.1. Algorithms Implemented

Bayesian Ridge Regressor

Bayesian regression has been one of the mechanisms that can help you survive unevenly distributed data by employing probability distributions instead of point estimates in linear regression. Rather than being estimated as a single value, the output 'y' is believed to be chosen from a probability distribution.

To obtain a completely probabilistic model, assume that y is Gaussian distributed around Xw X as follows.

$$p(y \mid X, w, \alpha) = N(y \mid X_w, \alpha)$$

One of the most effective types of Bayesian regression is the Bayesian Ridge Regressor, which estimates a probability model of the regression issue. The antecedent for the variable w is spherical in this case. Gaussian is denoted by the symbol

$$p(w \mid \lambda) = N\left(w \mid 0, \lambda^{-1} I_p\right)$$

```
[ ] from sklearn.model_selection import train_test_split
    from sklearn.metrics import r2_score
    from sklearn.linear_model import BayesianRidge

[ ] X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.15, random_state = 42)

    # Creating and training model
    model = BayesianRidge()
    model.fit(X_train, y_train)

    # Model making a prediction on test data
    prediction = model.predict(X_test)

    # Evaluation of r2 score of the model against the test set
    print(f"r2 Score Of Test Set : {r2_score(y_test, prediction)}")

    r2 Score Of Test Set : 0.6335204486315514

[ ] from sklearn import metrics
    print('Mean Absolute Error:', metrics.mean_absolute_error(y_test,prediction))
    print('Mean squared Error:', metrics.mean_squared_error(y_test,prediction))
    print('Root Mean Squared Error:', math.sqrt(metrics.mean_squared_error(y_test,prediction)))

    Mean Absolute Error: 0.5479611849903767
    Mean squared Error: 0.44788635816156275
    Root Mean Squared Error: 0.6692431233576949
```

Figure 11 Bayesian ridge regressor

2.7. Random Forest Regressor Using Hypertuning

The hyper parameters are similar to the parameters of algorithms that can be tweaked to improve performance. It entails determining a set of ideal hyper parameter values for the learning algorithm and then applying that algorithm to the dataset. It also said in the improvement of model performance and the production of better results.

Hyper parameters in a random forest include the handful of decision trees in the forest and the amount of attributes that each tree considers while splitting anode. The Random Forest Hyper parameters are as follows:

2.7.1. Maximum depth

In a random forest, it is made up of a long path between the node n and the leaf node. We can restrict the level of the tree in a random forest by using this option.

2.7.2. Split minimal samples

It is one of the parameters that inform the decision tree in a randomized forest how many observations are necessary in each node before it may be split.

The min sample split parameter is set to 2 by default. If a final node has far more than two

Observations, it can be divided into sub nodes, according to this rule.7

Leaf minimal samples

After splitting a node, this Randomized Forest hyper parameter sets the minimum amount of components that should be included in the leaf node.

Some of the advantages of using hyper parameters:

- Improves the accuracy of model and produce better results.
- Not changing the tree size (eg. max depth, min samples leaf, etc.) results in fully developed and un pruned trees, which on some data sets can be very enormous.

- Reducing memory consumption can be accomplished by altering the size and scope of the trees randomized forest using hyper tuning

```
[ ] from sklearn.model_selection import train_test_split
    X_train, X_test, y_train, y_test = train_test_split(X,Y,test_size = 0.2, random_state = 26)

[ ] n_estimators = [5,20,50,100] # number of trees in the random forest
    max_features = ['auto', 'sqrt'] # number of features in consideration at every split
    max_depth = [int(x) for x in np.linspace(10, 120, num = 12)] # maximum number of levels allowed in each decision tree
    min_samples_split = [2, 6, 10] # minimum sample number to split a node
    min_samples_leaf = [1, 3, 4] # minimum sample number that can be stored in a leaf node
    bootstrap = [True, False] # method used to sample data points

    random_grid = {'n_estimators': n_estimators,

    'max_features': max_features,

    'max_depth': max_depth,

    'min_samples_split': min_samples_split,

    'min_samples_leaf': min_samples_leaf,

    'bootstrap': bootstrap}

[ ] from sklearn.ensemble import RandomForestRegressor
    rf = RandomForestRegressor()
```

Figure 12 Random Forest Regressor Using Hyper tuning

2.8. Elasticnet Regressor

The algorithm used to describe the linearity between input data and the target output is known as linear regression. Elastic net Repressor is a linear regression extension that includes two penalty functions, L1 and L2. This entails including these penalties into the error function during training in order to encourage the use of simpler models with lower coefficient values. Normalized linear regression or penalized linear regression is the terms for these extensions.

Elastic net is a sort of regularized linear regression that includes two common penalty functions, the L1 and L2.

```
[ ] from sklearn import linear_model
    from sklearn.pipeline import Pipeline
    from sklearn.model_selection import GridSearchCV, cross_val_score
    from sklearn.preprocessing import StandardScaler
    from sklearn.metrics import mean_absolute_error
    from sklearn.decomposition import PCA

    xtrain, xtest, ytrain, ytest = train_test_split(X, Y, test_size=0.15)

[ ] std_slc = StandardScaler()

[ ] pca = PCA()

[ ] elasticnet = linear_model.ElasticNet()

[ ] pipe = Pipeline(steps=[('std_slc', std_slc),
                           ('pca', pca),
                           ('elasticnet', elasticnet)])
```

Figure 13a Elastic net regressor

```
[ ] print('Best Number Of Components:', clf_EN.best_estimator_.get_params()['pca__n_components'])
    print(clf_EN.best_estimator_.get_params()['elasticnet'])

    Best Number Of Components: 7
    ElasticNet(normalize=False, selection='random')

[ ] from sklearn.metrics import mean_squared_error


[ ] ypred = clf_EN.predict(xtest)
    score = clf_EN.score(xtest, ytest)
    mse = mean_squared_error(ytest, ypred)
    mae =mean_absolute_error(ytest,ypred)
    print("R2:{0:.3f}, MSE:{1:.2f}, RMSE:{2:.2f},MAE:{0:.3f}"
          .format(score, mse, np.sqrt(mse),mae ))

    R2:0.217, MSE:1.57, RMSE:1.25,MAE:0.217

[ ] myvals = np.array([24.59,0,1,1,0,4,4,3,0,5,0]).reshape(1,-1)
    p=clf_EN.predict(myvals)
    print("Predicted tip amount : ",math.ceil(p*100)/100)

    Predicted tip amount :  2.76
```

Figure 13b Elastic net regressor

3. Results and discussion

We compare RMSE, MSE, MAE values to find the accuracy of regress or algorithms these values should be minimum the less the values the more accurate the algorithm is. And R2 score should be high considering all these factors Bayesian ridge regress or have more accuracy. RMSE shows how better the model is able to fit the dataset.

Table 1 Accuracy values

	RMSE	MSE	MAE	R2	PREDICTED TIP
Random forest regressor using hyper tuning	0.83	0.69	0.62	0.44	3.22
Bayesian ridge regressor	0.66	0.44	0.54	0.633	2.62
Elasticnet regressor	1.25	1.57	0.21	0.217	2.76

4. Conclusion

By observing the results, we can say that for this dataset Bayesian Ridge Regressor predicted the output with almost the same accuracy (i.e., RMSE value=0.66) so, Bayesian ridge repressor has predicted the output as, it is the optimized technique its predicted value is almost similar to expected output.

By comparing restaurants with their rankings restaurant ‘a’ has highest ranking compared to other restaurants ‘b’, ‘c’, ‘d’ and the least rating for ‘b’ and also people gave more tips on weekends in dinner time with people count of 4. Who came for dinner rather than lunch and tip was given more for male who doesn’t smoke. We observe waiters prompt not only depends on waiters behavior but also restaurants ambience, customers finance i.e., if the customer is financially low then he cannot give more incentive to waiter even though waiters service is excellent. Grid is a tool that allows you to visualize the distribution of data of a single variable as well as the relationships between several variables within subsets of a dataset.

Compliance with ethical standards

Acknowledgments

We would like to express my gratitude to all the people behind the screen who helped me to transform an idea into a real application. We would like to express my heart-felt gratitude to my parents without whom we would not have been

privileged to achieve and fulfill my dreams. We are grateful to our CEO, Mr. K. Abhijit Rao, Director, Prof. C. V. Tomy, Principal, Dr. T. Ch. Siva Reddy**,** who most ably runs the institution and has had the major hand in enabling me to do my project. We profoundly thank Dr. Sunil Bhuthad**a**, Head of the Department of Information Technology who has been an excellent guide and also a great source of inspiration to my work. We would like to thank our Coordinator Dr. Subhani Shaik & internal guide Dr. K. Vijaya Lakshmi for their technical guidance, constant guidelines, encouragement and support in carrying out my project on time at college. The satisfaction and euphoria that accompany the successful completion of the task would be great but incomplete without the mention of the people who made it possible with their constant guidance and encouragement crowns all the efforts with success. In this context, we would like to thank all the other staff members, both teaching and non-teaching, who have extended their timely help and eased my task.

Disclosure of conflict of interest

No conflict of interest.

Statement of informed consent

No statement of informed consent.

References

[1] Azar, O. (2003). The implications of tipping for economic and management. International Journal of Socio-Economics, 30(10), 1084–1094

[2] Azar, O. (2004). The history of tipping from sixteenth-century England to United States in the 1910s. Journal of Socio-Economics, 33(6), 745-764.

[3] Ben-Zion, Uri and Edi Karni. 1977. Tip Payments and the Quality of Service. in O.C. Ashenfelter & W.E. Oates, (Eds.), Essays in Labor Market Analysis, 37- 44. New York: John Wiley & Sons.

[4] Bodvarson, O. & Gibson, W. (1994). Gratuities and customer appraisal of service: Evidence from Minnesota restaurants. Journal of Socio-Economics, 23(3), 287-302.

[5] Bodvarson, O. (2005). Restaurant tips and service quality: A reply to Lynn. Applied Economics Letters, 12, 345–346.

[6] Conlin, M., Lynn, M. & O'Donoghue, T. (2003). The norm of restaurant tipping. Journal of Economic Behavior and Organization, 52, 297-321.

[7] Garrity, K. & Degelman, D. (1990). Effect of server introduction on restaurant tipping. Journal of Applied Social Psychology, 20, 168-172.

[8] Incentive. (n.d.). In Merriam-Webster's online dictionary (11th ed.). Retrieved from 34 Koku, P.S. (2005). Is there a difference in tipping in restaurant versus non-restaurant service encounters, and do ethnicity and gender matter? Journal of Services Marketing, 19(7), 445-452.

[9] Lynn, M. & Graves, J. (1996). Tipping: An incentive/reward for service? Hospitality Research Journal, 20, 1-14.

[10] Lynn, M. & McCall, M. (2000). Gratitude and gratuity: A meta-analysis of research on the service-tipping relationship. Journal of Socio-Economics, 29, 203-214.

[11] Lynn, M. (2006). Tipping in restaurants and around the globe: An interdisciplinary review. Chapter 31, pp. 626-643. In M. Altman (Ed.) Handbook of contemporary behavioral Economics.

[12] Lynn, M. (2003). Tip levels and service: An update, extension and reconciliation. Cornell Hotel and Restaurant Administration Quarterly, 44, 139-148.

[13] Wessels, W. (1997) Minimum wages and tipped servers. Economic Inquiry 35(2), 334-49.

[14] Zeithaml, V.A., Bitner, M.J. & Gremler, D.,D. (2006). Services marketing: Integrating customer focus across the firm. New York: McGraw-Hill.