RCC Institute of Information Technology

INDUSTRIAL TRAINING PROJECT REPORT

ON THE TOPIC

'MACHINE LEARNING USING PYTHON'

AT WEBTEK LABS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

DEPARTMENT OF

ELECTRICAL ENGINEERING

SUBMITTED BY

MUSKAN MEHTA

ACKNOWLEDGEMENT:

It gives us great pleasure to acknowledge the guidance, assistance and support Ms. Mousita Dhar in making the Project and this Project report, which has been structured under her valued suggestion. She has helped us to accomplish the challenging task in a very short period of time.

Finally, We express the constant support of our college for inspiring us throughout and encouraging me.

We have undertaken industrial training at "WEBTEK LABS" during a period from 17 AUG to 10 SEPT in partial fulfillment of requirements for the award of degree of B.TECH (ELECTICAL ENGINEERING)
Beleaghata Kolkata, West Bengal 700152 under **MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY**. The work which is being presented in the training report submitted to Department of electrical engineering at (RCC INSTITUTE OF INFORMATION TECHNOLOGY is an authentic record of traning work.

The THREE WEEKS industrial training Viva–Voce Examination

of_____ has been held on _____ and accepted.

_____ _____

Signature of Internal Examiner Signature of External Examiner

CERTIFICATE OF APPROVAL :

The project "**WINE QUALITY PREDICTION**" made by MUSKAN METHA is hereby approved as a creditable study for the Bachelor of Technology in electrical engineering and presented in a manner of satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned this project only for the purpose for which it is submitted.

_____

Ms. Mousita Dhar
(Project In-charge)

INTRODUCTION :

❖ PYTHON :

➢ WHAT IS PYTHON ?

i. Python is a simple, easy to learn, powerful, high level and object-oriented programming language..

ii. Python is an interpreted scripting language also.

iii. Guido Van Rossum is known as the founder of python programming.

iv. Developers can read and translate Python code much easier than other languages.

v. The Python Software Foundation (PSF) is the organization behind Python.

➢ PYTHON VERSIONS :

- First released in 1991.
- Python 2.0 was released on 16 October 2000
- Python 3.0 was released on 3 December 2008
- Current Versions: i. 3.6.3
  ii. 2.7.14

➢ PYTHON FEATURES :

- ✓ Object oriented
- ✓ Easy to Learn and Use
- ✓ Interpreted Language
- ✓ Scripting Language
- ✓ Expressive Language
- ✓ Cross-platform Language
- ✓ GUI Programming Support
- ✓ Integrated Language
- ✓ Free and Open Source
- ✓ Extensible
- ✓ Large Standard Library

➢ APPLICATIONS OF PYTHON :

- **Web Applications :**

We can use Python to develop web applications. It provides libraries to handle internet protocols such as HTML and XML, JSON, Email processing, request, beautifulSoup, Feedparser etc. It also provides Frameworks such as Django, Pyramid, Flask etc to design and delelop web based applications. Some important developments are: PythonWikiEngines, Pocoo, PythonBlogSoftware etc.

- **Desktop GUI Applications :**

Python provides Tk GUI library to develop user interface in python based application. Some other useful toolkits wxWidgets, Kivy, pyqt that are useable on several platforms. The Kivy is popular for writing multitouch applications.

- **Software Development :**

Python is helpful for software development process. It works as a support language and can be used for build control and management, testing etc.

- **Scientific and Numeric :**

Python is popular and widely used in scientific and numeric computing. Some useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering, science and mathematics.

- **Business Applications :**

Python is used to build Bussiness applications like ERP and e-commerce systems. Tryton is a high level application platform.

- **Console Based Application :**

We can use Python to develop console based applications. For example: **IPython**.

- **Audio or Video based Applications :**

Python is awesome to perform multiple tasks and can be used to develop multimedia applications. Some of real applications are: TimPlayer, cplay etc.

- **3D CAD Applications :**

To create CAD application Fandango is a real application which provides full features of CAD.

- **Enterprise Applications :**

Python can be used to create applications which can be used within an Enterprise or an Organization. Some real time applications are: OpenErp, Tryton, Picalo etc.

- **Applications for Images :**

Using Python several application can be developed for image. Applications developed are: VPython, Gogh, imgSeek etc.

➢ **IDLE :**

IDLE is an integrated development environment for Python, which has been bundled with the default implementation of the language.

ANACONDA :

- Anaconda is a open source Distribution for data science and machine learning using python.
- Anaconda includes hundreds of popular data science packages and the *conda* package and virtual environment manager for Windows, Linux, and MacOS.
- Conda makes it quick and easy to install, run, and upgrade complex data science and machine learning environments like scikit-learn, TensorFlow, and SciPy.
- Anaconda Distribution is the foundation of millions of data science projects as well as Amazon Web Services
- *Machine Learning AMIs* and *Anaconda for Microsoft* on Azure and Windows.
- ANACONDA packages itself contain a IDE name SPYDER (Scientific Python Development Environment)

➢ Interactive Computation with IPython :

IPython is famous by the name of Jupyter Notebook these days. This gives you a freedom to write text between your code. Anaconda python packages by default contains it . Just go and launch IPython , After launching it, do not close the command prompt because it will work as a server. In few version, Ipython itself open a client tab on the default browser .In rest versions. The command window will give you an address with port.

➢ PACKAGES:

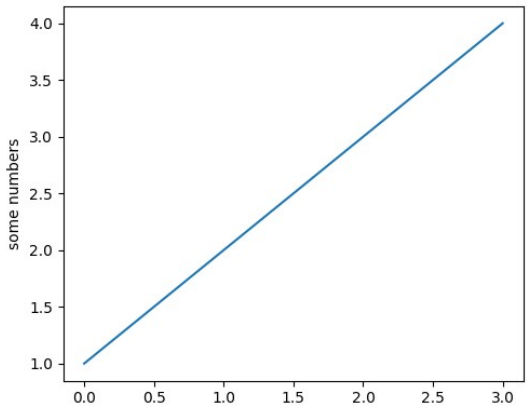
i. NumPy:

Numeric Python or NumPy is a Linear Alzebra Library for python. NumPy enriches the programming language Python with powerful data structures for efficient computation of multi-dimensional arrays and matrices. It also incredibly fast as it has bindings to C libraries.

ii. Matplotlib:

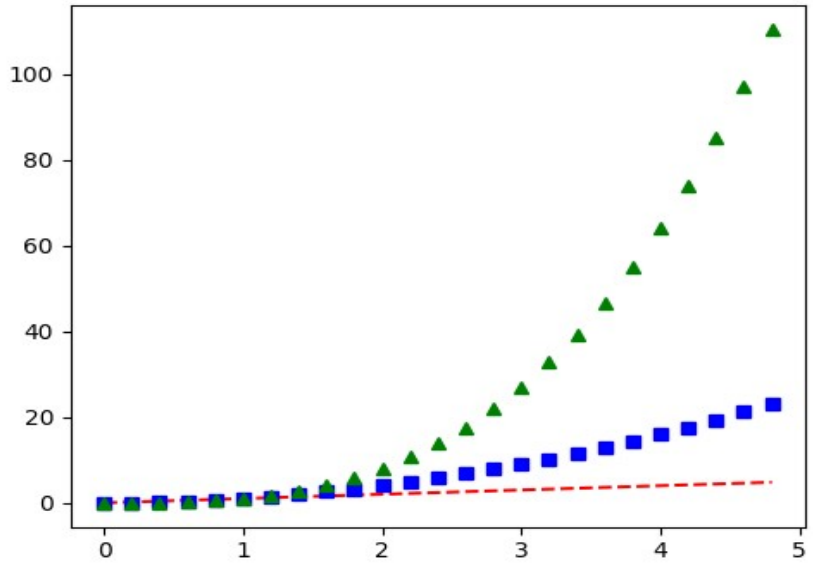
Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+.

```
>>> import matplotlib.pyplot as plt
    plt.plot([1,2,3,4])
    plt.ylabel('some numbers')
    plt.show()
```



plot() is a versatile command, and will take an arbitrary number of arguments.

If matplotlib were limited to working with lists, it would be fairly useless for numeric processing. Generally, you will use numpy arrays. In fact, all sequences are converted to numpy arrays internally. The example below illustrates a plotting several lines with different format styles in one command using arrays.



Matplotlib tries to make easy things easy and hard things possible. We can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc., with just a few lines of code. For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

iii. Scikit-learn :

Scikit-learn (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features
various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

iv. Pandas :

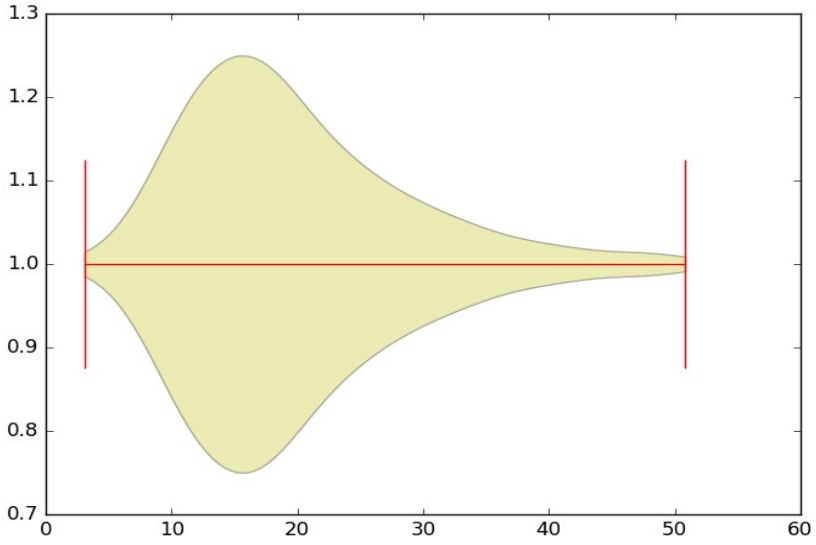
- Pandas is a Python module. It is a high performance, highly efficient, and high level data analysis library. Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. It has also a build-in visualization feature.
- Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data —load, prepare, manipulate, model, and analyze.
- CODE :

```
import pandas as pd
import datetime
import pandas_datareader.data as web
```

v. Seaborn :

- Seaborn or statistical data visualization. Seaborn is a Python visualization library based on matplotlib.
- It provides a high-level interface for drawing attractive statistical graphics.
- CODE :

```
# Import the necessary libraries
import matplotlib.pyplot as plt
import seaborn as sns

# Load the data
tips = sns.load_dataset("tips")

# Create violinplot
sns.violinplot(x = "total_bill", data=tips)
# Show the plot
plt.show()
```



MACHINE LEARNING :

❖ WHAT IS MACHINE LEARNING?

Machine learning is a data analytics technique that teaches computers to do what comes naturally to humans and animals: learn from experience. Machine learning algorithms use computational methods to "learn" information directly from data without relying on a predetermined equation as a model. The algorithms adaptively improve their performance as the number of samples available for learning increases.

❖ MACHINE LEARNING METHODS :

3 most famous categories are –

- SUPERVISED LEARNING
- UNSUPERVISED LEARNING
- REINFORCEMENT LEARNING

In these project we have used supervised learning methods. This algorithm consist of a target / outcome variable (or dependent variable) which is to be predicted from a given set of predictors (independent variables). Using these set of variables, we generate a function that map inputs to desired outputs. The training process continues until the model achieves a desired level of accuracy on the training data. Examples of Supervised Learning: Regression, Decision Tree, Random Forest, KNN, Logistic Regression etc.

❖ MACHINE LEARNING ALGORITHM :

There are several algorithm in machine learning. Like – logistic regression, linear regression, Decision Tree, Random Forest, SVM, KNN, K-means etc. In this project we have used RANDOM-FOREST algorithm.

➢ RANDOM-FOREST CLASSIFIER :

- Random Forest is a trademark term for an ensemble of decision trees. In Random Forest, we've collection of decision trees (so known as "Forest"). To classify a new object based on attributes, each tree gives a classification and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest).
- In a classification problem, each tree votes and the most popular class is chosen as the final result.
- In the case of regression, the average of all the tree outputs is considered as the final result.
- PROS :
  - ✓ Highly accurate and robust method.
  - ✓ Handle missing values
  - ✓ Used in both Classification and Regression
- CONS :
  - ✓ Random forests is slow in generating predictions because it has multiple decision trees.
  - ✓ The model is difficult to interpret compared to a decision tree, where you can easily make a decision by following the path in the tree.

PROJECT WORK :

❖ COLLECTING DATA FROM KAGGLE:

Kaggle is platform to compete with others in competitions which are based on machine learning tasks. It is a platform for predictive modeling and analytics competitions in which statisticians and data miners compete to produce the best models for predicting and describing the datasets uploaded by companies and users. This crowd sourcing approach relies on the fact that there are countless strategies that can be applied to any predictive modeling task and it is impossible to know beforehand which technique or analyst will be most effective.

On 8th March 2010, Google has announced that they have acquiring Kaggle. They will join the Google Cloud team and continue to be a distinct brand. In January 2018, Booz Allen and Kaggle launched Data Science Bowl, a machine learning competition to analyze cell images and identify nuclei.

❖ DATA SCIENCE :

Data science is a multidisciplinary blend of data inference, algorithm development, and technology in order to solve analytically complex problems. Data science is a "concept to unify statistics, data analysis, machine learning and their related methods" in order to "understand and analyze actual phenomena" with data. Turing award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational and now data-driven) and asserted that "everything
about science is changing because of the impact of information technology" and the data deluge.

When Harvard Business Review called it "The Sexiest Job of the 21st Century" the term became a buzzword, and is now often applied to business analytics, business intelligence, predictive modeling, or any arbitrary use of data, or used as a glamorized term for statistics. In many cases, earlier approaches and solutions are now simply rebranded as "data science" to be more attractive, which can cause the term to become "dilute beyond usefulness."

While many university programs now offer a data science degree, there exists no consensus on a definition or suitable curriculum contents. Because of the current popularity of this term, there are many "advocacy efforts" surrounding the field. To its discredit, however, many data science and big data projects fail to deliver useful results, often as a result of poor management and utilization of resources.

Here we have shown our dataset of “Wine_Quality” in below :

fixed acid	volatile ac	citric acid	residual s	chlorides	free sulfu	total sulfu	density	pH	sulphates	alcohol	quality
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5
7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8	5
11.2	0.28	0.56	1.9	0.075	17	60	0.998	3.16	0.58	9.8	6
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.4	0.66	0	1.8	0.075	13	40	0.9978	3.51	0.56	9.4	5
7.9	0.6	0.06	1.6	0.069	15	59	0.9964	3.3	0.46	9.4	5
7.3	0.65	0	1.2	0.065	15	21	0.9946	3.39	0.47	10	7
7.8	0.58	0.02	2	0.073	9	18	0.9968	3.36	0.57	9.5	7
7.5	0.5	0.36	6.1	0.071	17	102	0.9978	3.35	0.8	10.5	5
6.7	0.58	0.08	1.8	0.097	15	65	0.9959	3.28	0.54	9.2	5
7.5	0.5	0.36	6.1	0.071	17	102	0.9978	3.35	0.8	10.5	5
5.6	0.615	0	1.6	0.089	16	59	0.9943	3.58	0.52	9.9	5
7.8	0.61	0.29	1.6	0.114	9	29	0.9974	3.26	1.56	9.1	5
8.9	0.62	0.18	3.8	0.176	52	145	0.9986	3.16	0.88	9.2	5
8.9	0.62	0.19	3.9	0.17	51	148	0.9986	3.17	0.93	9.2	5
8.5	0.28	0.56	1.8	0.092	35	103	0.9969	3.3	0.75	10.5	7
8.1	0.56	0.28	1.7	0.368	16	56	0.9968	3.11	1.28	9.3	5
7.4	0.59	0.08	4.4	0.086	6	29	0.9974	3.38	0.5	9	4
7.9	0.32	0.51	1.8	0.341	17	56	0.9969	3.04	1.08	9.2	6
8.9	0.22	0.48	1.8	0.077	29	60	0.9968	3.39	0.53	9.4	6
7.6	0.39	0.31	2.3	0.082	23	71	0.9982	3.52	0.65	9.7	5
7.9	0.43	0.21	1.6	0.106	10	37	0.9966	3.17	0.91	9.5	5
8.5	0.49	0.11	2.3	0.084	9	67	0.9968	3.17	0.53	9.4	5

❖ SOURCE CODE & OUTPUT :

- **STEP 1 : Import Modules**

```
In [60]: import matplotlib.pyplot as plt
         import numpy as np
         import pandas as pd
         from sklearn.cross_validation import train_test_split
         from sklearn.metrics import accuracy_score
         import seaborn as sb
```

- **STEP 2 : Read Data From Dataset**

```
In [61]: #reading the data set
         data = pd.read_csv("C:\\Users\\Anuranjan\\Downloads\\winequality-red.csv")
```

```
In [62]: data.head()
```

Out[62]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

- **STEP 3 : Importing RandomForestClassifier**

```
In [63]: from sklearn.ensemble import RandomForestClassifier
```

- **STEP 4: Assigning X & Y axis**

```
In [64]: #assigning X and Y axis
         X = data.values[:,0:11]
         Y = data.values[:,11]
```

- **STEP 5: Splitting data into training set and test set**

```
In [65]: #splitting data into training set and test set
         X_train, X_test, y_train, y_test = train_test_split( X, Y, test_size = 0.25, random_state = 50)
```

- **STEP 6: Scaling Dataset,**

```
#scaling the data
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(X_train)
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
```

- **STEP 7 : Checking whether there is null value in the data set**

```
In [66]: #checking whether there is null value in the data set
         data.isnull().sum()
```

```
Out[66]: fixed acidity           0
         volatile acidity        0
         citric acid             0
         residual sugar          0
         chlorides               0
         free sulfur dioxide     0
         total sulfur dioxide    0
         density                 0
         pH                      0
         sulphates               0
         alcohol                 0
         quality                 0
         dtype: int64
```

- **STEP 8 : Assigning number of estimators (using decision tree)**

```
In [67]: #assigning number of estimators(decision trees used)
         clf=RandomForestClassifier(n_estimators=50,random_state = 40)
```

- **STEP 9 : Build the model**

```
In [68]: #using traing set data to build the model
         clf.fit(X_train,y_train)
```

```
Out[68]: RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                     max_depth=None, max_features='auto', max_leaf_nodes=None,
                     min_impurity_decrease=0.0, min_impurity_split=None,
                     min_samples_leaf=1, min_samples_split=2,
                     min_weight_fraction_leaf=0.0, n_estimators=50, n_jobs=1,
                     oob_score=False, random_state=None, verbose=0,
                     warm_start=False)
```

- **STEP 10 : Predict the data**

```
In [69]: #predicting the data
         y_pred=clf.predict(X_test)
```

```
In [70]: y_pred
```

```
Out[70]: array([ 6.,  6.,  6.,  6.,  5.,  5.,  5.,  5.,  5.,  5.,  5.,  5.,  6.,
                 6.,  6.,  5.,  5.,  6.,  5.,  6.,  5.,  6.,  5.,  6.,  5.,  5.,
                 5.,  5.,  5.,  5.,  6.,  5.,  6.,  5.,  5.,  6.,  7.,  6.,  5.,
                 5.,  6.,  5.,  6.,  6.,  5.,  6.,  5.,  5.,  5.,  6.,  6.,  7.,
                 6.,  5.,  5.,  5.,  6.,  5.,  6.,  5.,  5.,  5.,  6.,  6.,  5.,
                 5.,  5.,  6.,  6.,  5.,  6.,  6.,  6.,  6.,  5.,  6.,  5.,  5.,
                 6.,  5.,  5.,  4.,  7.,  5.,  5.,  5.,  5.,  3.,  6.,  5.,  5.,
                 5.,  5.,  7.,  5.,  7.,  7.,  6.,  6.,  5.,  5.,  5.,  6.,  6.,
                 5.,  5.,  6.,  6.,  5.,  5.,  5.,  6.,  6.,  5.,  6.,  5.,  5.,
                 6.,  6.,  5.,  6.,  5.,  6.,  5.,  6.,  6.,  5.,  6.,  5.,  5.,
                 5.,  6.,  5.,  5.,  5.,  6.,  3.,  5.,  5.,  5.,  5.,  5.,  6.,
                 5.,  7.,  5.,  5.,  6.,  7.,  6.,  6.,  5.,  6.,  5.,  6.,  5.,
                 5.,  5.,  5.,  6.,  5.,  5.,  5.,  5.,  7.,  6.,  7.,  6.,  5.,
                 6.,  6.,  6.,  5.,  5.,  5.,  5.,  5.,  5.,  5.,  6.,  5.,  6.,
                 5.,  5.,  5.,  5.,  7.,  5.,  5.,  7.,  6.,  5.,  7.,  6.,  6.,
                 6.,  5.,  6.,  6.,  6.,  6.,  7.,  7.,  5.,  7.,  6.,  5.,  5.,
                 5.,  5.,  5.,  5.,  6.,  5.,  7.,  5.,  6.,  6.,  6.,  5.,  7.,
                 7.,  5.,  6.,  5.,  5.,  5.,  5.,  6.,  6.,  6.,  6.,  6.,  5.,
                 5.,  6.,  6.,  6.,  5.,  6.,  6.,  5.,  6.,  6.,  5.,  5.,  6.,
                 6.,  6.,  6.,  6.,  5.,  5.,  6.,  6.,  4.,  5.,  5.,  6.,  6.,
                 5.,  5.,  6.,  6.,  6.,  5.,  5.,  5.,  6.,  6.,  5.,  6.,  6.,
                 6.,  7.,  5.,  6.,  5.,  6.,  5.,  6.,  5.,  5.,  5.,  5.,  5.,
                 6.,  5.,  5.,  5.,  6.,  5.,  5.,  6.,  6.,  5.,  5.,  6.,  7.,
                 6.,  6.,  5.,  6.,  5.,  5.,  6.,  7.,  6.,  5.,  6.,  7.,  5.,
                 7.,  5.,  6.,  5.,  5.,  6.,  5.,  6.,  6.,  6.,  5.,  6.,  6.,
                 5.,  6.,  6.,  7.,  5.,  5.,  5.,  6.,  5.,  5.,  5.,  6.,  8.,
```

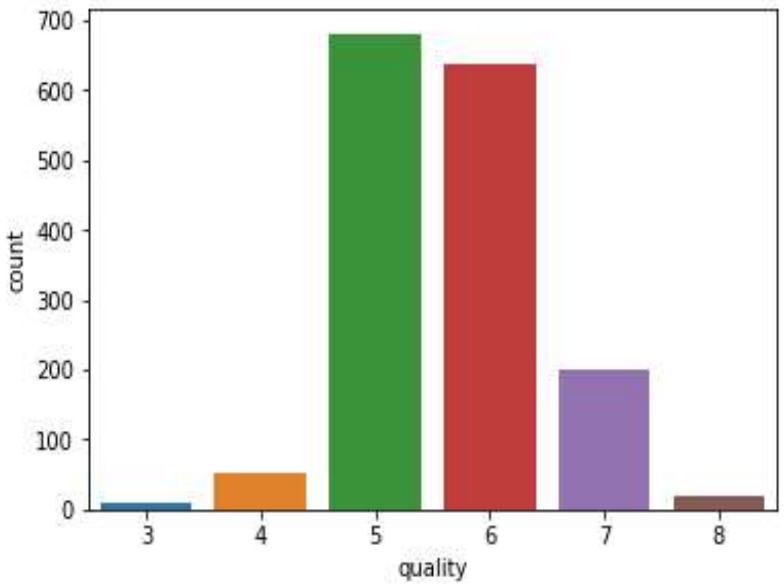
- **STEP 11 : Find the Score**

```
In [195]: #to find the score
          accuracy_score(y_test,y_pred)

Out[195]: 0.7275000000000004
```

- **STEP 12 : Plotting the bar chart between the quality and its count**

```
In [72]: #plotting the bar chart between the quality and its count
         sb.countplot(x='quality',data=data)

Out[72]: <matplotlib.axes._subplots.AxesSubplot at 0x2107c958748>
```

```
In [73]: plt.show()
```



❖ RESULTS AND CONCLUSION:

➢ RESULT:

By using Random-Forest Algorithm accuracy of around 72.75% can be achieved. Where as in SVM accuracy achieved is 60%.

So, we can say that our prediction in most cases will give the accurate result.

➢ CONCLUSION:

As we can see that our model is easy to implement and easy to implement. We designed a system to rate the quality of Wine. This system is capable of providing most of the essential features required to identify that which is more accurate quality of Wine. As we are all health conscious we have to choose the better one. We have done a small research on the wine quality. We do not do with a real dataset. So this method can be used in future research purposes. And the results can be used by the wine manufacturers to improve the quality of the future wines.

❖ FUTURE SCOPE:

We designed a system to rate the quality of Wine. This system is capable of providing most of the essential features required to identify that which is more accurate quality of Wine. As we are all health conscious we have to choose the better one. We have done a small research on the wine quality. We do not do with a

real dataset. So this method can be used in future research purposes. And the results can be used by the wine manufacturers to improve the quality of the future wines.

REFERENCES:

- https://en.wikipedia.org/wiki/Random_forest
- https://en.wikipedia.org/wiki/Wine_rating
- https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009
- https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/
- https://matplotlib.org/
- https://pandas.pydata.org/