Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 7, July : 2023

APPLYING DATA SCIENCE PRINCIPLES TO CLASSIFY EDIBLE OR POISONOUSMUSHROOMS

INDRANI BOBBURI, MCA, DCA, DVR & Dr.Hima Shekar MIC College of Technology, A.P., India.

K.MAHANTHI, Associate Professor, Dept.of AI & IT, DVR & Dr.Hima Shekar MIC college of Technology, A.P., India.

***Abstract*—**

Evaluation of the performance of various machine learning classification methods using a dataset of two different varieties of mushrooms and various mushroom characteristics.

This research project's primary goal is to use machine learning techniques to forecast the mushrooms in a dataset. Which mushrooms are toxic, or edible is predicted Machine learning regression.In fact, it is conceivable to predict the toxicity of mushrooms using machine learning, and this idea has been investigated in the mycology community. A collection of mushroom attributes and labels indicating whether each mushroom is harmful, or edible can be used to train machine learning systems.It's crucial to remember that even if machine learning might make useful predictions, it shouldn't take the place of mycology experts' knowledge. It might be difficult to distinguish between edible and deadly mushrooms, and doing so can have detrimental effects. Therefore, if dealing with mushroom toxicity, always seek the advice of a professional or turn to trustworthy sources.

INTRODUCTION

Scientists and mathematicians started investigating the concept of artificial intelligence in the early 1900s. One of the concepts under examination is machine learning, which teaches a machine how to think and behave like a human. One such subject is data science, which involves drawing conclusions from numerous, huge datasets. Artificial intelligence (AI) is a combination of machine learning, data science, and machine learning. Recommendations, Classification, Pattern Detection, Prediction, Grouping, Anomaly Detection, Recognition, and Forecasting are just a handful of the numerous potential deliverables from data analysis.

The measurements and characteristics of the mushrooms are used in this study to categories them as either edible or poisonous. As a result, this exercise involves binary classification. The data is classified using four Machine Learning techniques.

The following sections comprise the remaining portions of this essay. The data is described in the second part, along with its feature set and output classes. The technique, including data preparation and the tools used to characterise the data, are described in the third part. Each algorithm's modelling classification performance is assessed and reported in the fourth portion, and conclusions are reached in the fifth and final section

.A diverse collection of fungus known as mushrooms come in a wide range of colours, sizes, and shapes. While certain mushrooms can be ingested and used in dishes, others can be extremely toxic and provide a serious health concern. In mycology, determining which mushrooms are toxic and which are edible is a crucial task that frequently calls for specialised knowledge. But with machine learning improvements, we may use data-driven strategies to build prediction models that help in precisely classifying mushrooms.

In this project, we want to use Python to create a machine learning model that can predict how hazardous mushrooms would be. We are able to identify patterns and features that distinguish between hazardous and edible mushrooms by training a model on a dataset of labelled mushroom samples. The trained model will subsequently be able to categorise fresh, undiscovered samples of mushrooms according to their characteristics, offering useful information to researchers, foragers, and mushroom enthusiasts.

1. Data collection: We will compile a vast database of mushroom samples, each of which will have labels indicating whether it is toxic or edible. Our machine learning model will be trained and tested using this dataset as its basis. reliable sources, such as the UCI Machine Learning Repository or American colleges.

2. Data Preprocessing: We will carry out a number of preprocessing operations to get the data ready for analysis. In doing so, it may be necessary to handle missing values, transform category features into numerical representations, and, if necessary, normalise or standardise the data. During this phase, careful consideration will be given to ensuring the dataset's integrity.

3. Feature Selection/Extraction: It is essential to recognise the pertinent aspects that support the categorization task. Using data analysis, we will choose useful traits that can distinguish between poisonous and edible mushrooms. We'll take into account characteristics that are frequently employed in mushroom categorization, including cap form, cap colour, gill size, and odour.

4. Model Selection: Accurate prediction requires selecting the right machine learning method. In this section, we'll examine a number of classification algorithms, including decision trees, logistic regression, support vector machines (SVM), random forests, and gradient boosting algorithms like XGBoost or LightGBM. The size, complexity,

and criteria for interpretability of the dataset will all have an impact on the approach used.

5. Training the model: The dataset will be divided into training and testing sets. The chosen machine learning model will be trained using the training set on the mushroom features and associated labels. The model will figure out the underlying patterns in the data during training, which will allow it to make precise predictions.

6. Model Evaluation: Using the right evaluation criteria, we will evaluate the trained model's performance. We will use metrics like accuracy, precision, recall, and F1 score to gauge how well the model can identify if a mushroom is toxic or edible. To confirm the model's generalizability, this evaluation will use the testing set, which consists ofunseen data.

7. Model Optimization: To improve the performance of the model, it may be essential to adjust the model's parameters and investigate various feature engineering or ensemble methods. To optimise the model's configuration and raise the model's predicted precision, cross-validation and hyperparameter tuning will be used.

8. Predictions: After the model has been trained and assessed, we will use it to forecast the toxicity of fresh samples of mushrooms. The model will output the probability or binary classification indicating whether the mushroom is harmful or edible after users enter the attributes of the fungus.

LITERATURE REVIEW

A literature survey of mushroom data analysis involves reviewing and summarizing existing research articles, publications, and studies related to the analysis of mushroom data. This type of survey aims to provide a comprehensive understanding of the current state of knowledge, research trends, methodologies, and findings in the field of mushroom data analysis.

The literature survey typically begins with identifying relevant keywords, such as "mushroom data analysis," "fungi data mining," "mushroom classification," or "mushroom pattern recognition." These keywords are used to search various academic databases, journals, conference proceedings, and other reputable sources to gather relevant articles.

In the survey, the selected articles are thoroughly read, and the main findings, methodologies, and conclusions are extracted and summarized. The survey may cover various aspects of mushroom data analysis, including but not limited to:

1. Mushroom classification: This involves the development of classification models to distinguish edible and poisonous mushrooms or to classify different mushroom species based on their characteristics.

2. Feature extraction and selection: Identifying relevant features or attributes that best represent the characteristics of mushrooms for analysis and decision-making processes.

3. Pattern recognition: Analyzing patterns and trends in mushroom data to identify correlations, associations, or recurring characteristics.

4. Data mining techniques: Applying data mining algorithms, such as decision trees, support vector machines, or neural networks, to extract useful information from mushroom datasets.

5. Quality assessment: Evaluating the reliability and accuracy of the collected mushroom data and developing quality assessment methods to ensure data integrity.

6. Disease prediction and management: Analyzing mushroom data to predict and prevent diseases or optimize disease management strategies in mushroom cultivation.

RELATED WORK

The objective of a literature survey on mushroom data analysis is to provide a comprehensive overview of the current state of knowledge, research trends, methodologies, and findings in the field of analyzing mushroom data. The survey aims to achieve the following objectives:

1. Identify Existing Research: The literature survey aims to identify and review existing research articles, publications, and studies related to mushroom data analysis. This helps in understanding the existing body of knowledge and the research efforts that have been made in the field.

2. Summarize Findings and Methodologies: The survey involves thoroughly reading and summarizing the main findings, methodologies, and conclusions of the selected articles. This provides a concise summary of the existing research and helps in understanding the various approaches and techniques used for analyzing mushroom data.

3. Identify Research Gaps: By reviewing the literature, the survey aims to identify any gaps or limitations in the existing research. This helps in identifying areas where further research is needed or where improvements can be made.

4. Highlight Trends and Emerging Techniques: The survey helps in identifying emerging trends and techniques in mushroom data analysis. It provides insights into the latest developments in the field, such as new algorithms, tools, or methodologies that researchers are exploring.

5. Provide Recommendations: Based on the findings of the literature survey, recommendations can be provided for improving data analysis techniques, expanding data sources, or exploring novel approaches to mushroom data analysis. These

recommendations can guide future research efforts and help in advancing the field.

https://scikit-

6. Support Decision-Making and Planning: The literature survey provides a knowledge base that can support decision-making and planning for researchers, practitioners, or stakeholders involved in mushroom data analysis. It helps in understanding the current landscape and the potential applications and challenges in the field.

Overall, the objective of a literature survey on mushroom data analysis is to gather and synthesize existing knowledge, identify gaps, and provide insights and recommendations to advance the field of mushroom data analysis.

PROPOSED WORK

1. Dataset:
  - UCI Machine Learning Repository: Mushroom Data Set (https://archive.ics.uci.edu/ml/datasets/mushroom)

2. Python Libraries:
  - Pandas Documentation: https://pandas.pydata.org/docs/
  - Scikit-learn Documentation: https://scikit-learn.org/stable/documentation.html
  - NumPy Documentation: https://numpy.org/doc/

3. Machine Learning Algorithms:
  - Logistic Regression:
    - Scikit-learn LogisticRegression Documentation:

learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

- Support Vector Machines (SVM):
  - Scikit-learn SVM Documentation: https://scikit-learn.org/stable/modules/svm.html

- XGBoost:
  - XGBoost Documentation: https://xgboost.readthedocs.io/

- LightGBM:
  - LightGBM Documentation: https://lightgbm.readthedocs.io/

- Naïve Bayes:
  - Scikit-learn Naive Bayes Documentation: https://scikit- learn.org/stable/modules/naive_bayes.html

4. Data Science Tutorials and Guides:

- Kaggle: https://www.kaggle.com/
  - Kaggle is a platform that hosts various data science projects, including mushroom classification. You can find datasets, kernels (code notebooks), and discussions related to mushroom analysis.

- Towards Data Science: https://towardsdatascience.com/
  - Towards Data Science is an online publication that offers a wide range of articles, tutorials, and case studies related to data science and machine

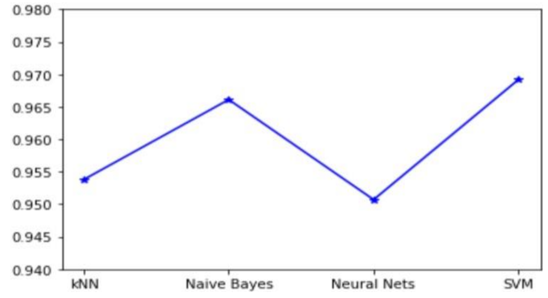
learning. You can find articles on various aspects of mushroom analysis and classification.

- Machine Learning Mastery: https://machinelearningmastery.com/

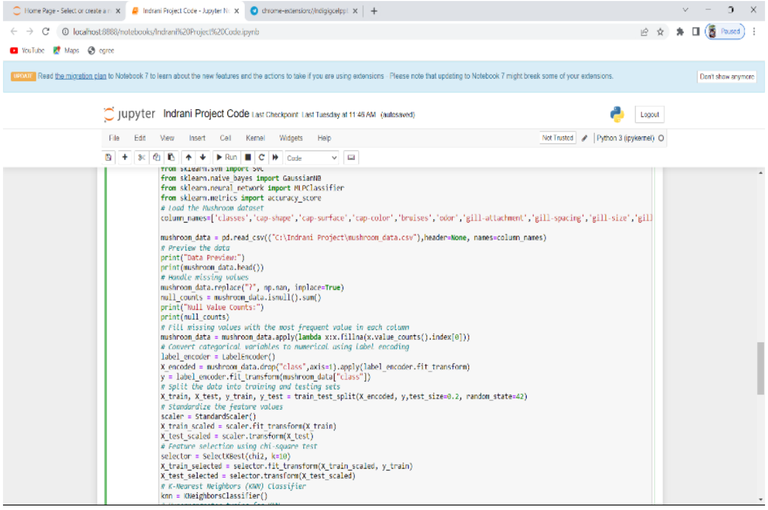
- Machine Learning Mastery is a website that provides tutorials, guides, and resources on machine learning algorithms and techniques. It offers articles related to classification tasks and specific algorithms such as logistic regression, SVM, XGBoost, and Naïve Bayes.

Remember to properly cite any references you use in your project, especially if you directly use code, datasets, or ideas from external sources.

```
[<matplotlib.lines.Line2D at 0x13afbed10>]
```

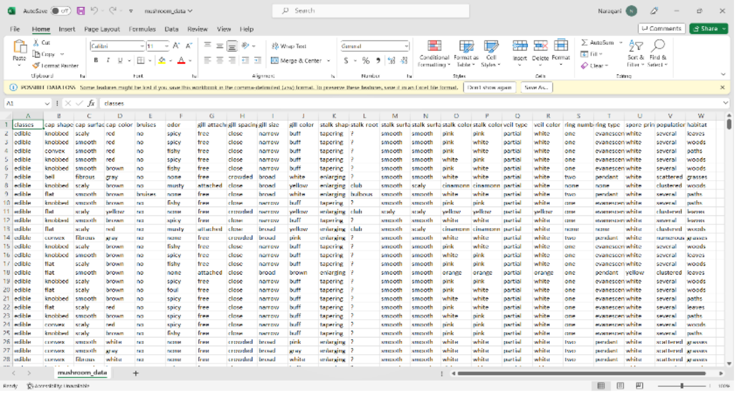


SAMPLE SCREENSHOTS

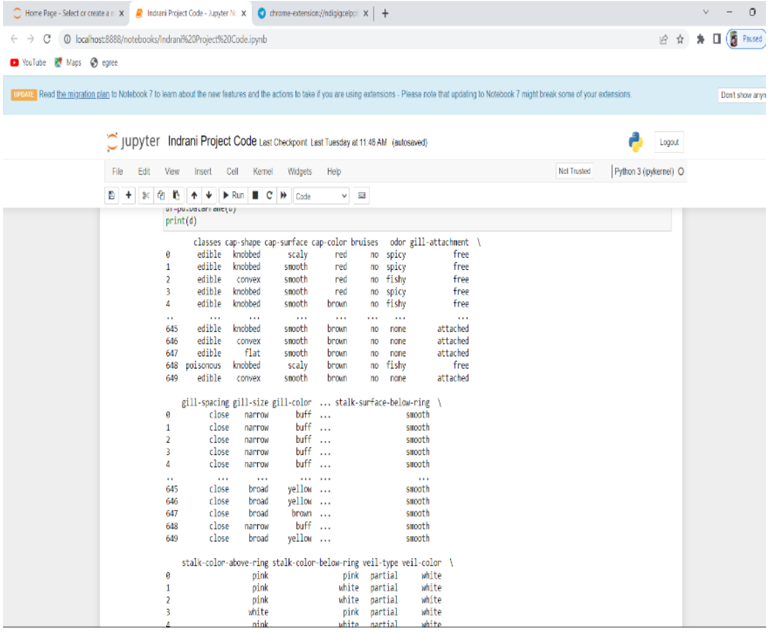

localhost:8888/notebooks/Indrani%20Project%20Code.ipynb
Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions · Please note that updating to Notebook 7 might break some of your extensions.
Don't show anymore
jupyter
Indrani Project Code Last Checkpoint: Last Tuesday at 11:46 AM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code

```
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score
# Load the Mushroom dataset
column_names=['classes','cap-shape','cap-surface','cap-color','bruises','odor','gill-attachment','gill-spacing','gill-size','gill

mushroom_data = pd.read_csv(("C:\Indrani Project\mushroom_data.csv"),header=None, names=column_names)
# Preview the data
print("Data Preview:")
print(mushroom_data.head())
# Handle missing values
mushroom_data.replace("?", np.nan, inplace=True)
null_counts = mushroom_data.isnull().sum()
print("Null Value Counts:")
print(null_counts)
# Fill missing values with the most frequent value in each column
mushroom_data = mushroom_data.apply(lambda x:x.fillna(x.value_counts().index[0]))
# Convert categorical variables to numerical using label encoding
label_encoder = LabelEncoder()
X_encoded = mushroom_data.drop("class",axis=1).apply(label_encoder.fit_transform)
y = label_encoder.fit_transform(mushroom_data["class"])
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_encoded, y,test_size=0.2, random_state=42)
# Standardize the feature values
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
# Feature selection using chi-square test
selector = SelectKBest(chi2, k=10)
X_train_selected = selector.fit_transform(X_train_scaled, y_train)
X_test_selected = selector.transform(X_test_scaled)
# K-Nearest Neighbors (KNN) Classifier
knn = KNeighborsClassifier()
```



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	classes	cap shape	cap surfac	cap color	bruises	odor	gill attach	gill spacing	gill size	gill color	stalk shap	stalk root	stalk surfa	stalk surfa	stalk color	stalk color	veil type	veil color	ring numb	ring type	spore prin	population	habitat
2	edible	knobbed	scaly	red	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	leaves
3	edible	knobbed	smooth	red	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	woods
4	edible	convex	smooth	red	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	paths
5	edible	knobbed	smooth	red	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	white	pink	partial	white	one	evanescen	white	several	woods
6	edible	knobbed	smooth	brown	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	woods
7	edible	bell	fibrous	gray	no	none	free	crowded	broad	white	enlarging	?	smooth	smooth	white	white	partial	white	two	pendant	white	scattered	grasses
8	edible	knobbed	scaly	brown	no	musty	attached	close	broad	yellow	enlarging	club	smooth	scaly	cinamonn	cinamonn	partial	white	none	none	white	clustered	woods
9	edible	flat	smooth	brown	bruises	none	free	close	broad	white	enlarging	bulbous	smooth	smooth	white	white	partial	white	two	pendant	white	several	paths
10	edible	knobbed	smooth	brown	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	paths
11	edible	flat	scaly	yellow	no	none	free	crowded	narrow	yellow	enlarging	club	scaly	scaly	yellow	yellow	partial	yellow	one	evanescen	white	clustered	leaves
12	edible	knobbed	smooth	brown	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	white	white	partial	white	one	evanescen	white	several	leaves
13	edible	flat	scaly	red	no	musty	attached	close	broad	yellow	enlarging	club	smooth	scaly	cinamonn	cinamonn	partial	white	none	none	white	clustered	woods
14	edible	convex	fibrous	gray	no	none	free	crowded	broad	pink	enlarging	?	smooth	smooth	white	white	partial	white	two	pendant	white	numerous	grasses
15	edible	knobbed	scaly	brown	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	woods
16	edible	knobbed	smooth	brown	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	white	pink	partial	white	one	evanescen	white	several	leaves
17	edible	flat	scaly	brown	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	woods
18	edible	flat	smooth	brown	no	none	attached	close	broad	brown	enlarging	?	smooth	smooth	orange	orange	partial	orange	one	pendant	yellow	clustered	leaves
19	edible	knobbed	scaly	brown	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	woods
20	edible	flat	scaly	brown	no	foul	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	woods
21	edible	knobbed	smooth	brown	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	white	white	partial	white	one	evanescen	white	several	paths
22	edible	flat	scaly	red	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	white	partial	white	one	evanescen	white	several	leaves
23	edible	knobbed	smooth	brown	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	white	pink	partial	white	one	evanescen	white	several	paths
24	edible	convex	scaly	red	no	spicy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	woods
25	edible	knobbed	scaly	brown	no	fishy	free	close	narrow	buff	tapering	?	smooth	smooth	pink	pink	partial	white	one	evanescen	white	several	paths
26	edible	convex	smooth	white	no	none	free	crowded	broad	pink	enlarging	?	smooth	smooth	white	white	partial	white	two	pendant	white	scattered	grasses
27	edible	convex	smooth	gray	no	none	free	crowded	broad	gray	enlarging	?	smooth	smooth	white	white	partial	white	two	pendant	white	scattered	grasses
28	edible	convex	fibrous	white	no	none	free	crowded	broad	white	enlarging	?	smooth	smooth	white	white	partial	white	two	pendant	white	scattered	grasses


Home Page - Select or create a n
Indrani Project Code - Jupyter N
chrome-extension://ndigigcelppl
localhost:8888/notebooks/Indrani%20Project%20Code.ipynb
Paused
YouTube
Maps
egree
UPDATE
Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions.
Don't show anym
jupyter
Indrani Project Code Last Checkpoint: Last Tuesday at 11:48 AM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code

```
df=pd.DataFrame(d)
print(d)
```

```
       classes cap-shape cap-surface cap-color bruises   odor gill-attachment  \
0       edible   knobbed       scaly       red      no  spicy            free
1       edible   knobbed      smooth       red      no  spicy            free
2       edible    convex      smooth       red      no  fishy            free
3       edible   knobbed      smooth       red      no  spicy            free
4       edible   knobbed      smooth     brown      no  fishy            free
..         ...       ...         ...       ...     ...    ...             ...
645     edible   knobbed      smooth     brown      no   none        attached
646     edible    convex      smooth     brown      no   none        attached
647     edible      flat      smooth     brown      no   none        attached
648  poisonous   knobbed       scaly     brown      no  fishy            free
649     edible    convex      smooth     brown      no   none        attached

    gill-spacing gill-size gill-color  ... stalk-surface-below-ring  \
0          close    narrow       buff  ...                   smooth
1          close    narrow       buff  ...                   smooth
2          close    narrow       buff  ...                   smooth
3          close    narrow       buff  ...                   smooth
4          close    narrow       buff  ...                   smooth
..           ...       ...        ...  ...                      ...
645        close     broad     yellow  ...                   smooth
646        close     broad     yellow  ...                   smooth
647        close     broad      brown  ...                   smooth
648        close    narrow       buff  ...                   smooth
649        close     broad     yellow  ...                   smooth

    stalk-color-above-ring stalk-color-below-ring veil-type veil-color  \
0                     pink                   pink   partial      white
1                     pink                  white   partial      white
2                     pink                  white   partial      white
3                    white                   pink   partial      white
4                     pink                  white   partial      white
```

CONCLUSION

In this paper a systematic evaluation was carried out of the performance of four different classification algorithms to classify mushrooms types using the features/properties. It was shown that all four classification algorithms performed very well (all above 95%) and are therefore well suited to this classification problem. SVM achieved the highest overall percentage of correctly classified instances of 96.92%.

REFERENCES:

James.G., Witten.D, Hastie.T.,Tibshirani.R.,(2017) An Introduction to Statistical Learning , with Applications in R . 2nd Edition. Springer

Simundić AM. Bias in research. Biochem Med. 2013;23:12–5. http://dx.doi.org/10.11613/BM.2013.003. [PMC free article] [PubMed] [Google Scholar]

Ray Kurzweil, *The Age of Spiritual Machines: When Computers Exceed Human Intelligence*Viking | 0-670-88217-8

Gene Bylinsky, *Computers That Learn By Doing*, Fortune, September 6, 2019.

Dennis Collins, *BrainMaker: Strange, Captivating, Easy to Use*, CaliforniaComputer News, July,2020.

AUTHOR PROFILES:

MR. K.MAHANTHI completed M.S. He has published 1 paper in JES journal. Currently working as an M.S.Assistant professor in the department of AI and IT at DVR & DR. HSMIC College of Technology (Autonomous), Kanchikacherla, NTR(DT). His areas of interest include C language, Data science and Python,Web technologies.

MS. INDRANI BOBBURI is MCA Student in the department of Computer Applications at DVR & DR.HS MIC College of Techonology(Automonous), Kanchikacherla, NTR(DT). She has Completed B.Sc in SRSVRGNRCollegeofArts & Science , Mylavaram, NTR (DT). Her areas of interests are Machine Learning, Web Technologies, Python and HTML