Blind Modulation Identification Using Machine Learning And Deep Learning Algorithms

G.Krishna Murthy[1], **M.Mani Kumar Reddy**[2]

[1,2]Assistant Professor (Adhoc), Department of ECE, JNTUA College of Engineering, Pulivendula.

Abstract:

Blind Identification of Modulations is One important physical-layer signal processing technique in wireless communication networks is automated modulation categorization (AMC), which improves spectrum utilisation by blindly identifying the modulation type of incoming signals at the receiver. The many high-impact accomplishments in radio signal processing for communications and other informatics domains driven by deep learning (DL) and ML have inspired the development of several modern AMC techniques using deep networks to address the limitations of earlier methods. Machine learning has the potential to enhance modulation classification performance when dealing with channel flaws by learning radio signal properties for pattern identification of modulation. Deep Learning, Decision Tree, Bagging, and KNN are some of the classifiers that may be used to categorise digital modulation signals of higher order. A popular choice for many applications, the Decision Tree classifier is noted for its interpretability, simplicity, and practicality. Its accuracy success rate of 82% is quite impressive. Still, when compared to other machine learning algorithms, Deep Learning outperforms them all with an accuracy of 92%. This illustrates the ability of Deep Learning models, in particular neural networks, to handle the intricate patterns and complex relationships present in the dataset.

Keywords: Automated Modulation Classification, Deep Learning, Machine Learning, Decision Tree, Wireless Signal Processing.

1. Introduction

Autonomous radio spectrum comprehension is necessary for military electronic warfare and threat assessments. Applications such as spectrum interference detection, monitoring, and dynamic spectrum access require it in civil contexts [1].Communication networks have historically relied on intricate digital and analogue modulation techniques to achieve a balance between spectrum economy and transmission reliability [8]. To handle high traffic in massive wireless communication systems, however, excessive spectrum consumption in highly-linked networks has led to co-channel interference and distortion of signals across propagation channels. Intelligent spectrum management is made possible by the non-cooperative configurations used by current communication systems which encode radio signals using modulation forms chosen from a pre-defined candidate pool based on system parameters and channel circumstances. Modulation identification algorithms are often given priority in software-defined radio-based communications because they allow the receiver to demodulate incoming signals by automatically recognising their forms of modulation [10][18]. The signal processing and communication sectors are showing increasing interest in automatic modulation categorization (AMC), the step that comes before physical layer signal demodulation [6]. The AMC approach classifies the modulation type of the signal that arrives at the receiver in order to handle a multi-class decision-making challenge connected to AI. The extraction of features and feature selection are prominent feature engineering approaches that may be used to retrieve basic radio properties like modulation type [7]. After that, a classification model may be constructed utilising supervised or unsupervised learning with these characteristics. AMC has a number of additional challenges, including growing modulation formats, significant channel limitations, and intra-class differentiation of higher-order digital modulations [8] digital modulations.

Transmission signals in analog communication systems are encoded using analog modulations such as AM, PM, and FM.[11] [15]. Analog modulation is used to encode the source signal, which is an analog baseband signal, into a periodic waveform at a high frequency. Because they work in tandem with digital data and are resistant to interference, digital modulations are superior to their analog counterparts. Digital modulation communication methods first sample and quantize the source signal. Then, in order to improve security and reduce transmission mistakes, the signal is coded. At the end, they send it to the digital modulator. Some common examples of digital modulations are pulse amplitude modulation (PAM), amplitude and phase-shift keying (APSK), and quadrature amplitude modulation (QAM)[23][5][6]. The properties of the carrier signal waveform, such as its amplitude, frequency, phase, or a mix of the two, may be altered using the pre-defined modulation method. In order to determine the optimal way to modulate an incoming signal along propagation channels, a learned AI model is used to deduce its radio characteristics.We focus on the advantages and disadvantages of the most popular ML and statistics models. The contributions to the research publication are streamlined [20]. As a result, readers will be able to pick out the essential features of each method [2][13].

The use of deep learning (DL) for image and voice recognition has been helpful [2]. By sequentially adding CNN, LSTM, and DNN layers, the CLDNN model is able to combine the best features of these networks into a single, more effective model.To distinguish between various kinds of modulation, CNN, ResNet, and LSTM are used in channel simulations. Compared to traditional techniques, DL-based systems perform better [1].[6].This application often makes use of decision trees, neural networks, and KNNs as classifiers. Machine learning algorithms create statistical data patterns for classification after selecting signal input qualities. Making use of Radio ML 2016.10A, we achieve state-of-the-art classification [19]. The main motto is to find out efficient classifier that can predict the modulation type of the given input signal automatically with a good accuracy using both ml classifiers and deep learning classifier. The objective of this paper is to take Four ML classifiers and a deep learning classifier. they are trained and tested on the dataset for finding out effective classifier out of them by comparing the accuracy of the classifiers.

2. Literature survey

- In 1998, Nandi introduced a Decision Theory-based approach that utilized parameters derived from Instantaneous Amplitude & Phase and Instantaneous Frequency. The modulation types examined included AM, DSB, SSB, VSB, FM, LSB, and USB. The developed techniques were tested on a variety of band-limited analog and digitally modulated signals that were tainted by band-limited Gaussian noise sequences using computer simulations.[5]
- Wu's 2005 work concentrated on a Feature-Based approach employing a Neural Tree Network. It incorporated power spectral features and Cyclic Spectra features of the IF signal, covering modulation types such as CW, FSK2, FSK4, FSK8, BPSK, QPSK, and QAM8. Existing Feature-Based methods were applied at a fixed SNR level, highlighting the challenge when SNR levels vary, necessitating potential re-training of classifiers for different channel environments.[13]
- In 2011, Ataollah proposed an optimized Radial Basis Function Neural Network (RBFNN) for Automatic Recognition of Communication Signals (ARCS). The method emphasized 4th order moments, addressing modulation types like ASK2, ASK4, and PSK2. The hybrid system comprised three main modules: feature extraction, classifier, and optimization modules. The classifier module featured a radial basis function neural network, and the optimization module utilized genetic algorithms to optimize recognizer design, proving effective primarily for digital modulation signals.[17]
- Mohannad, in 2018, developed an automatic modulation classification technique utilizing moments and likelihood maximization, specifically focusing on the 4th and 6th order moments. The suggested method, which relies on statistical moments in conjunction with a maximum likelihood

engine, constituted a hybrid approach at the connection of likelihood-based and feature-based classifiers and was applicable to BPSK, QPSK, and QAM64.[21]

3. System model

Figure 1 shows the suggested approach. It includes training and testing. To evaluate performance, a total of a thousand samples of each modulation class are generated under different noise conditions. Depending on the training rate parameter, some of the data is utilised for training and the rest for testing[16].

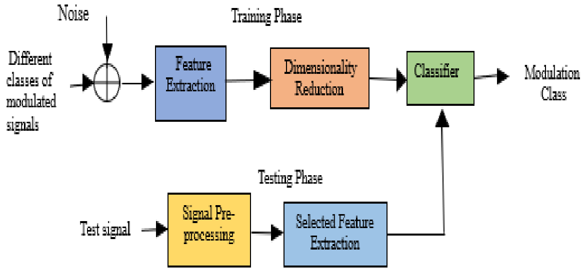


Figure 1: System Model Block Diagram

3.1 Signal model:

The received signal of adaptive modulation, r(t) is given by:

$$r(t) = s(t) + n(t)$$

Noise is added to the initial signal S(t) by the additive white Gaussian noise (AWGN) channel., which is represented by n(t). Convention dictates that in-phase and quadrature components be used to represent raw data for feature calculation and analysis [17].

$$a[i] = a_I[i] + j * a_Q[i]$$

Because of this, signals exhibit both real and imaginary components that correspond to the constellation diagrams. This model calculates and represents all statistical features and constellations. In the forward pass of the generic CNN, each filter is convolved over the input, as

$$y[n] = x[n] * h[n] = \sum_{l=0}^{L-1} x[l].h[n-1]$$

Where the convolutional layer output is denoted by y[n] and the incoming input is represented by x[n]. The L-filter is denoted by h[n].There is another important metric for representing signals and noise in the actual world and it is described by: signal-to-noise ratio (SNR).

$$SNR = \frac{Power\ of\ x\ signal}{x\ variance\ of\ noise} = 20\log_{10}\frac{\sqrt{\frac{1}{N}\sum|a[i]|^2}}{std(|noise|)}$$

Transmissions ranging from -20 to 18 dB SNR are the focus of this research. Verifying that the SNR is less than 0 dB is essential for discriminating between different types of modulation. Our models were built for a whole dataset, spanning from -20 to 18 dB SNR [17][24].

3.2 Feature extraction:

Make improvements to machine learning algorithms' classification accuracy and pattern discovery capabilities relative to their raw data training by adding features to existing data.Features are created by feature engineering using raw data. Recognizing modulation often makes use of cyclic-moment characteristics [2]. This notebook has thirty-two unique features. There is no cyclic time lag used by the first sixteen features; the latter sixteen make use of eight. Here are some features:

$s_{mn} = f_m(\{x_i^n x_{i+T}^n\})$.

mth order statistic on the nth power of the received signal x_i, whether it is instantaneous or delayed. A dual-stream AMC structure is created using CNN-LSTM [2]. Two streams are involved: one learns from phase and amplitude data, while the other takes local temporal characteristics from raw signals. To learn both geographical and temporal information from streams, CNN-LSTM integrates CNN's spatial feature extraction with LSTM's time-series processing capabilities [6]. By enabling features learned from two streams to interact in pairs, effective operations improve signal classification and increase the diversity of characteristics[3][16].

4. Data set:

Here Radio ML 2016.10a dataset is used to assess several machine learning architectures. The Radio ML 2016.10a dataset contains 220,000 labelled In-Phase samples and 220,000 labelled Quadrature samples, both of which are examples of two-dimensional data. Each synthetic dataset has eleven modulations, eight digital and three analog, and was built using GNU Radio. A variety of signal-to-noise ratio (SNR) settings, including light fading, minor Local Oscillators (LO) drift, and numerous marked SNR increments, are tested with the synthetic signals.[17]

- Signals that are modulated digitally: 8PSK, BPSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK [10]
- Signals that are modulated analogly: AM-DSB, AMSSB, and WBFM

5. K-nearest neighbors model

Regression and classification are two common applications for the supervised learning method known as the K Nearest Neighbour approach [2]. This flexible method may also be applied to dataset resampling and missing value imputing. As the name suggests, K Nearest Neighbour predicts the class or continuous value of a new data point by using the data points that are nearest to it.In terms of classification, the new data point's predicted class is the one to which the majority of the training dataset's K Nearest have been allocated [4][6].

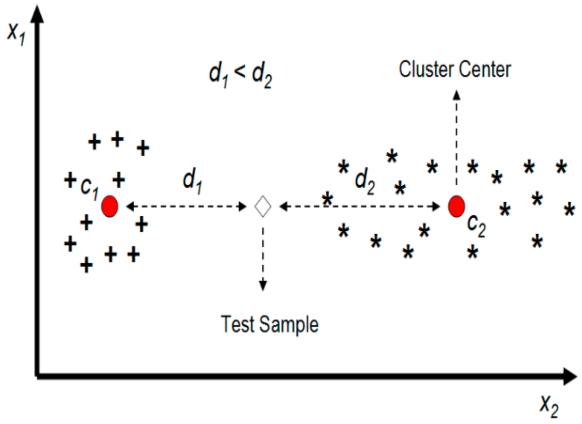


Figure: 2 KNN Model Diagram

The KNN algorithm finds the closest points or groups for a query point. To get the closest groups or points for a query point, require Euclidean Distance, Manhattan Distance, etc [9][25].

5.1 Distance metrics used in knn model:

- **Euclidean Distance:** SKlearn's default K-Nearest Neighbor measure is this distance, making it the most popular. It measures the real straight line distance between two Euclidean locations.Set Minkowski stance metric up to 2 to use it[8].

$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

- **Manhattan Distance:** This distance is called taxicab distance or city block distance because of its calculation. The absolute differences of two Cartesian coordinates determine their distance[8].The Manhattan distance formula is obtained by replacing p=1 in the Minkowski distance calculation.

d = $\sum_{i=1}^{n} |x_i - y_i|$

- **Minkowski Distance:** For vector spaces with real values, this measure is applicable. Only in a normed vector space, where distances may be expressed as non-negative vectors, can the Minkowski distance be computed [9].Distance metrics must meet certain conditions:

The function d(x, y) is non-negative if and only if it >= 0.

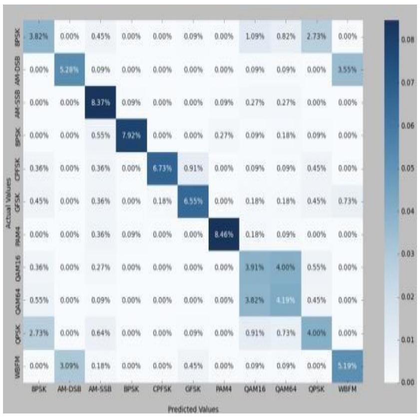
1. The identity is that d(x, y) = 0 only when x == y.
2. A symmetric function is defined as d(x, y) = d(y, x). 6.
3.Triangle Inequality: d(x, y) + d(y, z) >= d(x, z)

$$\left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p}$$

- The Manhattan distance is obtained when p is set to 1.
- We get the Euclidean distance when p is set to 2.

5.2 Heatmap for the knn model

Heatmap is a color-coded data visualization of matrix values[12]. This uses reddish colors to symbolize common values or higher activities and darker colors to represent less common or lower activities. The shading matrix name defines heatmap.



	8PSK	AM-DSB	AM-SSB	BPSK	CPFSK	GFSK	PAM4	QAM16	QAM64	QPSK	WBFM
8PSK	3.82%	0.00%	0.45%	0.00%	0.00%	0.09%	0.00%	1.09%	0.82%	2.73%	0.00%
AM-DSB	0.00%	5.28%	0.09%	0.00%	0.00%	0.00%	0.00%	0.09%	0.09%	0.00%	3.55%
AM-SSB	0.00%	0.00%	8.37%	0.09%	0.00%	0.00%	0.09%	0.27%	0.27%	0.00%	0.00%
BPSK	0.00%	0.00%	0.55%	7.92%	0.00%	0.00%	0.27%	0.09%	0.18%	0.09%	0.00%
CPFSK	0.36%	0.00%	0.36%	0.00%	6.73%	0.91%	0.00%	0.09%	0.09%	0.45%	0.00%
GFSK	0.45%	0.00%	0.36%	0.00%	0.18%	6.55%	0.00%	0.18%	0.18%	0.45%	0.73%
PAM4	0.00%	0.00%	0.36%	0.09%	0.00%	0.00%	8.46%	0.18%	0.09%	0.00%	0.00%
QAM16	0.36%	0.00%	0.27%	0.00%	0.00%	0.00%	0.00%	3.91%	4.00%	0.55%	0.00%
QAM64	0.55%	0.00%	0.09%	0.00%	0.00%	0.00%	0.00%	3.82%	4.19%	0.45%	0.00%
QPSK	2.73%	0.00%	0.64%	0.00%	0.00%	0.09%	0.00%	0.91%	0.73%	4.00%	0.00%
WBFM	0.00%	3.09%	0.18%	0.00%	0.00%	0.45%	0.00%	0.09%	0.09%	0.00%	5.19%

Figure 3: Confusion Matrix of the Modulated classes

6. Decision tree clssifier model

To classify and predict outcomes, decision trees use non-parametric supervised learning techniques. The nodes that make up a hierarchical tree are the root, branches, internal, and leaf nodes. Simple regression and classification models are provided by decision trees [14].

Choices and their likely outcomes, including random occurrences, resource costs, and utility, are shown using decision support hierarchical models, such as decision trees. Conditional control statements are used for classification and regression in this non-parametric supervised learning algorithmic technique [13]. Each level of a hierarchical tree is represented by a different kind of node: the root, branches, internal, and leaf.

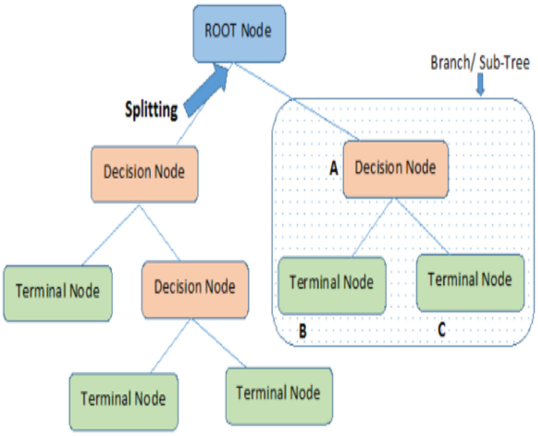


Figure 4: Decision Tree Model

6.1 Attribute selection measures:

Using the attribute selection measure (ASM), decision tree algorithms examine attributes in order to partition datasets. Data may be more effectively divided into homogeneous subgroups using ASM, which strives to maximise information acquisition [5].

Two well-known approaches to ASM are:

1. Knowledge Acquired
2. Index of Gini

Information gain:

Claude Shannon devised entropy to quantify input impurity. Entropy is a system's randomness or impurity in physics and arithmetic. An impurity in a set of instances in information theory. Information acquisition reduces entropy [5][6]. The information gain is the change from the initial and average entropy before and after attribute-based dataset splitting. An method called ID3 decision tree takes use of data gain [8].

Info(D) = $- \sum_{i=1}^{m} p_i \log_2 p_i$

The chance that a random D tuple belongs to class Ci is Pi.

$Info_A$(D) = $\sum_{j=1}^{V} \frac{|D_J|}{|D|}$ $X\ Info\ (D_j)$

Gain (A) = Info(D) – $Info_A(D)$

Where:

- Info(D) is the average amount of data needed to identify a tuple in D's class label.
- |Dj|/|D indicates the weight of the jth partition
- The purpose of InfoA(D) is to classify a tuple from D according to A partitioning. Gain(A), the splitting attribute at node N, has the highest information gain.

Entropy:

Sample impurity is measured by entropy. If all S samples are in one class, Entropy is 0. The best-case situation represents purity. Entropy is one if half the samples are in one class and half in another. The worst-case scenario. Splitting on the attribute with the lowest entropy is preferable. Entropy measures divided priority.

$H(s) = - p_+ \log_2 p_+ - p_- \log_2 p_-$

Where, p_- - probability of positive class

p_+ - probability of negative class

Gini index:

In statistical and financial situations, the Gini Index measures impurity or inequality. Impurity measures are used in decision tree algorithms for classification problems in machine learning [13]. The Gini Index ranges from 0 (absolutely pure) to 1 (perfectly impure) and evaluates the chance of a decision tree algorithm misclassifying a randomly selected test.

Gini Index formula: $\text{Gini Index} = 1 - \sum_j P_j^2$

7. Bagging classifier model

Ensemble metaestimators like bagging classifiers fit basic classifiers to random subsets of the dataset, and then they produce a final prediction by voting or averaging their predictions [12]. By randomly constructing an ensemble, a metaestimator may reduce the variance of a black-box estimator (such a decision tree).

"Bagging" refers to the process of training several base models simultaneously using different subsets of the training data. Bootstrap sampling generates subjects by selecting and replacing data points at random. The final all-base model prediction of the bagging classifier is based on majority voting [12][26].

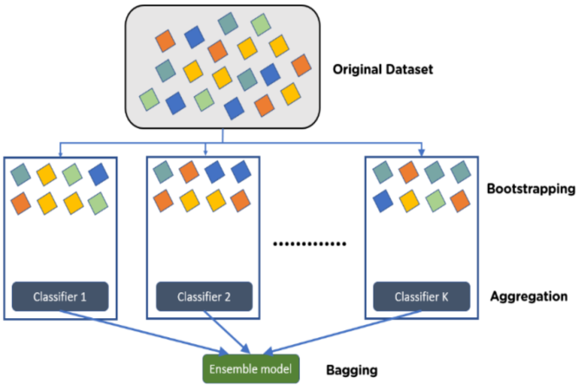


Figure 5: Bagging Classifier

8. Pasting classifier model:

Pasting is an algorithm that draws random sections of the dataset as samples. Similar to Bagging, Pasting samples training data subsets without replacement. This ensures that subsets do not include duplicate data points by selecting each data point once [12].

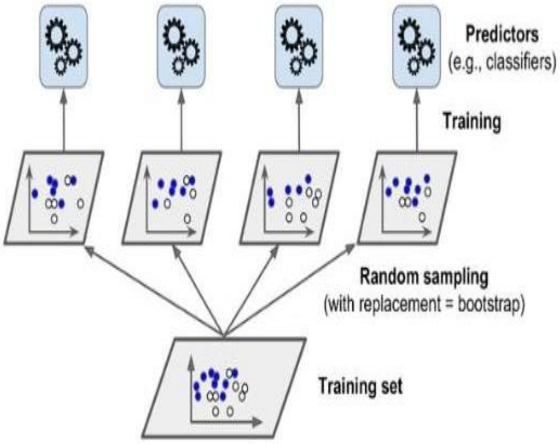


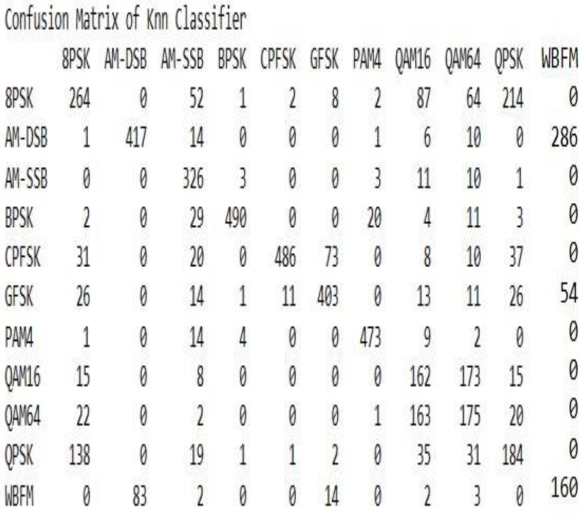
Figure 6: Pasting Classifier

9. Simulation results and discussions

Over SNR levels from -20db to 18dB, 11 modulated signals are used to evaluate four classifiers. From each modulated signal, 32 characteristics from feature selection are recovered to determine modulation classes. After then, the 110000*32 feature set was split into training and testing sets. The performance analysis began with 90% of the feature set as training data and 10% as testing data.

The assessment of recommended K-nearest neighbors (KNN) classifiers for Body Mass Index (BMI) is performed under less-than-ideal channel conditions. The simulation encompasses signals 8PSK, BPSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK, AM-DSB, AM-SSB, and WBFM to gauge the efficiency of the proposed BMI classifiers. Diverse frequency characteristics and statistical features are considered across Signal-to-Noise Ratio (SNR) values spanning from -20 to 18 dB.

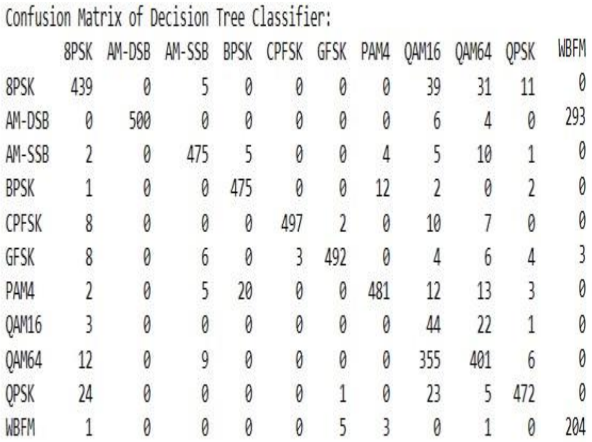
Examine 1000 examples in each modulation class and subject them to different noise levels throughout the experimental trials. Eleven distinct statistical characteristics are collected for each instance of a modulated signal, meeting training and testing criteria. Performance is divided into training (90%) and testing (10%) sets for assessment purposes. The proposed BMI's main performance metrics evaluate the true positive rate, or classification accuracy, across a variety of Signal-to-Noise Ratio (SNR) values.As a result, the KNN Classifier's acc uracy at 18dB is 64%.



```
Confusion Matrix of Knn Classifier
       8PSK  AM-DSB  AM-SSB  BPSK  CPFSK  GFSK  PAM4  QAM16  QAM64  QPSK  WBFM
8PSK    264       0      52     1      2     8     2     87     64   214     0
AM-DSB    1     417      14     0      0     0     1      6     10     0   286
AM-SSB    0       0     326     3      0     0     3     11     10     1     0
BPSK      2       0      29   490      0     0    20      4     11     3     0
CPFSK    31       0      20     0    486    73     0      8     10    37     0
GFSK     26       0      14     1     11   403     0     13     11    26    54
PAM4      1       0      14     4      0     0   473      9      2     0     0
QAM16    15       0       8     0      0     0     0    162    173    15     0
QAM64    22       0       2     0      0     0     1    163    175    20     0
QPSK    138       0      19     1      1     2     0     35     31   184     0
WBFM      0      83       2     0      0    14     0      2      3     0   160
```

Figure 7: Confusion Matrix of KNN Model

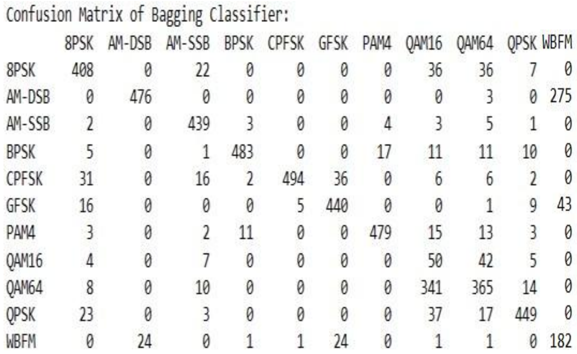
A decision tree classifier, which is renowned for its ease of use and minimal complexity, was employed in an effort to discriminate between eight digital modulation signals and three analogue signals. In the process of training the decision tree classifier, parameters are optimised through the use of randomised search. The classifier has a maximum of 133 leaf nodes and a maximum depth of 13. Entropy is the classifier's criterion. 82% accuracy was attained by the Decision Tree Classifier with a signal-to-noise ratio (SNR) of 18 dB.



```
Confusion Matrix of Decision Tree Classifier:
        8PSK  AM-DSB  AM-SSB  BPSK  CPFSK  GFSK  PAM4  QAM16  QAM64  QPSK  WBFM
8PSK     439       0       5     0      0     0     0     39     31    11     0
AM-DSB     0     500       0     0      0     0     0      6      4     0   293
AM-SSB     2       0     475     5      0     0     4      5     10     1     0
BPSK       1       0       0   475      0     0    12      2      0     2     0
CPFSK      8       0       0     0    497     2     0     10      7     0     0
GFSK       8       0       6     0      3   492     0      4      6     4     3
PAM4       2       0       5    20      0     0   481     12     13     3     0
QAM16      3       0       0     0      0     0     0     44     22     1     0
QAM64     12       0       9     0      0     0     0    355    401     6     0
QPSK      24       0       0     0      0     1     0     23      5   472     0
WBFM       1       0       0     0      0     5     3      0      1     0   204
```

Figure 8: Confusion Matrix of Decision Tree Model

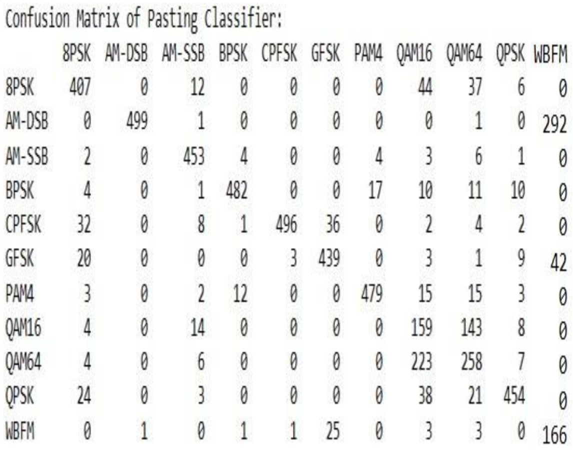
The Bagging Classifier Algorithm functions as an ensemble technique aimed at enhancing accuracy compared to a single classifier. It has proven to be notably more impactful than individual classifiers, elevating prediction outcomes through the amalgamation of multiple base estimators. Operating as an extension of Decision Tree Classifiers, this classifier has demonstrated a noteworthy accuracy rate of 79%, surpassing the performance of the KNN classifier.



```
Confusion Matrix of Bagging Classifier:
        8PSK  AM-DSB  AM-SSB  BPSK  CPFSK  GFSK  PAM4  QAM16  QAM64  QPSK WBFM
8PSK     408       0      22     0      0     0     0     36     36     7    0
AM-DSB     0     476       0     0      0     0     0      0      3     0  275
AM-SSB     2       0     439     3      0     0     4      3      5     1    0
BPSK       5       0       1   483      0     0    17     11     11    10    0
CPFSK     31       0      16     2    494    36     0      6      6     2    0
GFSK      16       0       0     0      5   440     0      0      1     9   43
PAM4       3       0       2    11      0     0   479     15     13     3    0
QAM16      4       0       7     0      0     0     0     50     42     5    0
QAM64      8       0      10     0      0     0     0    341    365    14    0
QPSK      23       0       3     0      0     0     0     37     17   449    0
WBFM       0      24       0     1      1    24     0      1      1     0  182
```

Figure 9: Confusion Matrix of Bagging Classifier

The Pasting Classifier, a form of ensemble learning closely linked to the Bagging Classifier, differentiates itself by utilizing random sampling without replacement. In contrast to Bagging, Pasting follows this distinctive approach. The classifier exhibits an accuracy level of 78% at a Signal-to-Noise Ratio (SNR) of 18dB.



```
Confusion Matrix of Pasting Classifier:
        8PSK  AM-DSB  AM-SSB  BPSK  CPFSK  GFSK  PAM4  QAM16  QAM64  QPSK  WBFM
8PSK     407       0      12     0      0     0     0     44     37     6     0
AM-DSB     0     499       1     0      0     0     0      0      1     0   292
AM-SSB     2       0     453     4      0     0     4      3      6     1     0
BPSK       4       0       1   482      0     0    17     10     11    10     0
CPFSK     32       0       8     1    496    36     0      2      4     2     0
GFSK      20       0       0     0      3   439     0      3      1     9    42
PAM4       3       0       2    12      0     0   479     15     15     3     0
QAM16      4       0      14     0      0     0     0    159    143     8     0
QAM64      4       0       6     0      0     0     0    223    258     7     0
QPSK      24       0       3     0      0     0     0     38     21   454     0
WBFM       0       1       0     1      1    25     0      3      3     0   166
```

Figure 10: Confusion Matrix of Pasting Classifier

VISUALIZATIONS OF CLASSIFIERS PERFORMANCE:

The graphs below show the performance of the various classifiers across different SNR levels. Among the Classifiers tested, KNN achieved 65% accuracy, decision tree 82%, bagging 79%, and pasting 78% Accuracy, all while operating at an SNR level of 18 dB.

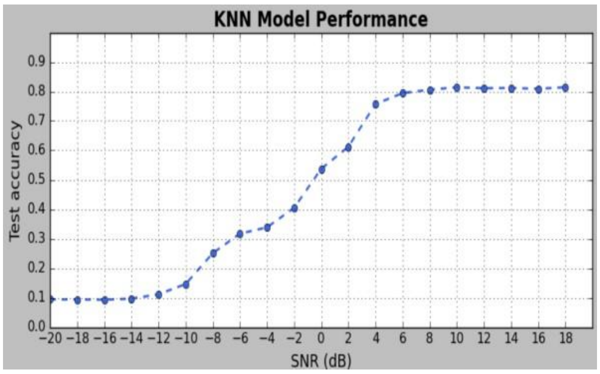


Figure 11 : Accuracy Plot for KNN Model

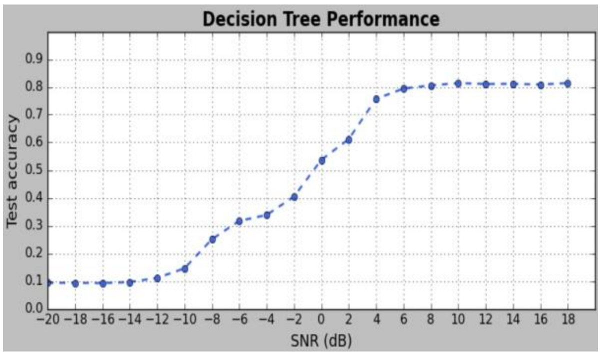


Figure 12: Accuracy Plot for Decision Tree Model

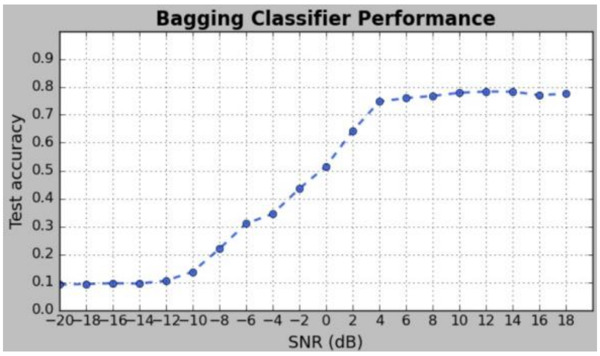


Figure 13: Accuracy Plot for Bagging Model

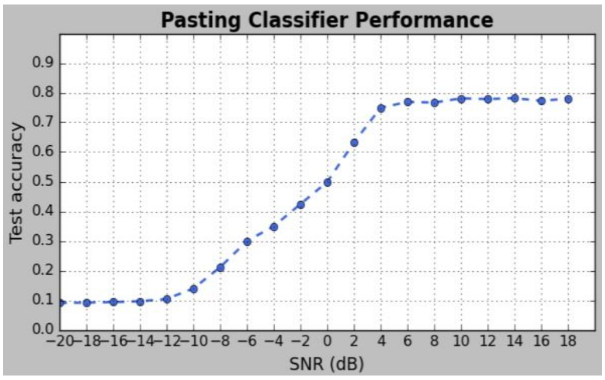


Figure 14: Accuracy Plot for Pasting Model

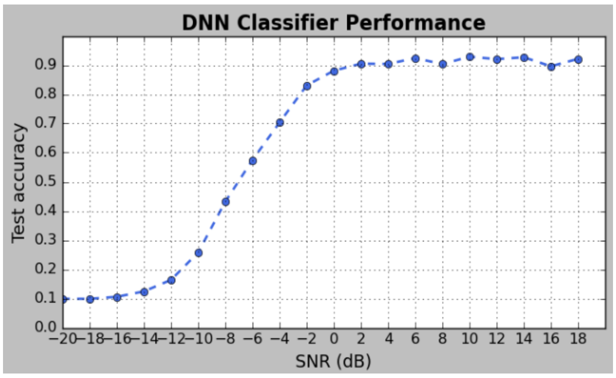


Figure 15: Accurcay plot of DNN Model

COMPARISON GRAPH FOR DIFFERENT CLASSIFIERS:

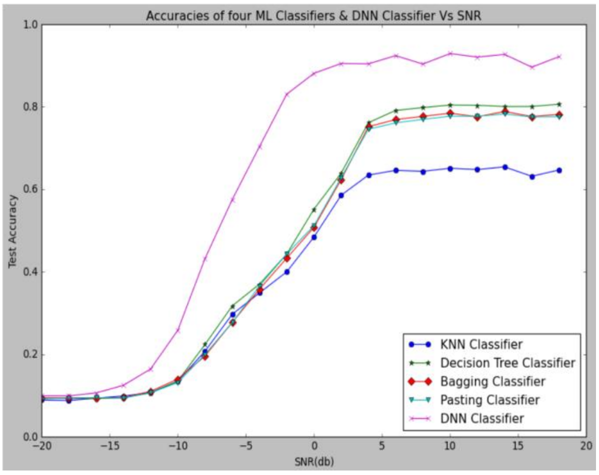


Figure 16: Comparision Graph

When comparing the accuracy performance of four different classifiers—Decision Tree, KNN, Bagging, and Pasting it become clear that Decision Tree performs the best. With an accuracy rate of 82%, the Decision Tree classifier outperforms the competition. Bagging and Pasting classifiers also shown almost relatively same performance as Decision Tree Classifiers with an accuracy of 79%. However, when compared to ML methods, Deep Learning yields the best accuracy of 92%. Therefore, for Blind Modulation Identification, Decision Tree Classifier is proven to be efficient classifier among other Machine Learning models and Deep Learning is more efficient than Decision tree Classifier.

10. Conclusion

Classification of modulated signals, whether digital or analog, is addressed in this work using a variety of machine learning and deeplearning approaches. The 32 most essential characteristics were recovered from the RadioML dataset by feature extraction. Each model is trained using its own dataset, which has been divided into a training dataset and a testing dataset. here several model's accuracy is tested and evaluated its performance on the test dataset.. Deep Learning outperforms the Machine Learning and its competitors in terms of accuracy. Because of this, it has mostly replaced Machine Learning methods.

REFERENCES:

1. "O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, “Survey of automatic modulation classification techniques: classical approaches and new trends,” IET Communications, vol. 1, no. 2, pp. 137–156, 2007"
2. "Multitask-Learning-Based Deep Neural Network for Automatic Modulation Classification Shuo Chang, Sai Huang, Ruiyun Zhang, Zhiyong Feng, Liang Liu"
3. "G. Q. Yang, “Modulation classification based on extensible neural networks,” Mathematical Problems in Engineering, vol. 9, p. 6416019, 2017."
4. "Automatic Modulation Classification Using Gated Recurrent Residual Network Sai Huang,Rui Dai,Juanjuan Huang,Yuanyuan Yao,Yue Gao,Fan Ning,Zhiyong Feng
5. K. Nandi, and E. E Azzouz, “Algorithms for Automatic Modulation Recognition of Communication Signals,” IEE [11] E Trans. Commun., vol. 46, pp. 431-436, 1998
6. Abu-Romoh, M., Aboutaleb, A., Rezki, Z. (2018).Automatic modulation classification using moments and likelihood maximization. IEEE Communications Letters, 22(5): 938-941. https://doi.org/10.1109/LCOMM.2018.2806489.

7. Hu, S., Pei, Y., Liang, P.P., Liang, Y. C. (2019). Deep neural network for robust modulation classification under uncertain noise conditions. IEEE Transactions on Vehicular Technology, 69(1): 564-577. "https://doi.org/10.1109/TVT.2019.2951594"
8. Wu, H.C., Saquib, M., Yun, Z. (2008). Novel automatic modulation classification using cumulant features for communications via multipath channels. IEEE Transactions on Wireless Communications, 7(8): 3098- 3105. "https://doi.org/10.1109/TWC.2008.070015"
9. Subbarao, M.V., Samundiswary, P. (2019). Knearest neighbors based automatic modulation classifier for next generation adaptive radio systems. Int J. Security Appl., 13: 41-50.
10. Venkata Subbarao, M., and P. Samundiswary. "Spectrum sensing in cognitive radio networks using time– frequency analysis and modulation recognition." Microelectronics, Electromagnetics and Telecommunications. Springer, Singapore, "https://doi.org/10.1007/978-981-10-7329-8_85".
11. Panagiotou, P., Anastasopoulos, A., Polydoros, A. (2000). Likelihood ratio tests for modulation classification. In MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority (Cat. No. 00CH37155), 2, pp. 670674. https://doi.org/10.1109/MILCOM.2000.904013
12. Wu, Z., Zhou, S., Yin, Z., Ma, B., Yang, Z. (2017). Robust automatic modulation classification under varying noise conditions. IEEE Access, https://doi.org/10.1109/ACCESS.2017.2746140
13. Wu, Y., & Li, J. (2005). Comprehensive features based digital modulation identification using a neural tree network. *Proceedings of the 2005 International Conference on Communications, Circuits and Systems.* DOI: 10.1109/ICCCAS.2005.1495219
14. "A. Afan and Y. Y. Fan, “Unsupervised feature learning and automatic modulation classification using deep learning model,” Physical Communication, vol. 25, pp. 75–84, 2017.
15. Dobre, O.A., Hameed, F. (2006). Likelihoodbased algorithms for linear digital modulation classification in fading channels. In 2006 Canadian conference on electrical and computer engineering, pp. 1347-1350. IEEE. https://doi.org/10.1109/CCECE.2006.277525
16. Zhang, Z., Hua, Z., Liu, Y. (2017). Modulation classification in multipath fading channels using sixthorder cumulants and stacked convolutional autoencoders. IET Communications, 11(6): 910-915. https://doi.org/10.1049/iet-com.2016.0533
17. Ataollah, M. (2011). Using Optimized Radial Basis Function Neural Network and Feature-Based 4th Order Moments for Automatic Modulation Classification. In Proceedings of the 2011 IEEE International Conference on Communications, Circuits, and Systems (pp. 24). DOI: 10.1109/ICCCAS.2011.6137815.
18. Subbarao, M.V., Samundiswary, P. (2018). Automatic modulation recognition in cognitive radio receivers using multi-order cumulants and decision trees. Int. J. Rec. Technol. Eng.(IJRTE), 7: 61-69.
19. Venkata Subbarao, M., Samundiswary, P. (2021). Automatic modulation classification using cumulants and ensemble classifiers. In Advances in VLSI, Signal Processing, Power Electronics, IoT, Communication and Embedded Systems, pp. 109-120. https://doi.org/10.1007/978-981-16-0443-0_9
20. Wong, M.D., Nandi, A.K. (2006). Blind phaseamplitude modulation classification with unknown phase offset. In 18th International Conference on Pattern Recognition (ICPR'06), pp. 177-180. https://doi.org/10.1109/ICPR.2006.333
21. Mohannad A. R. (2018). Automatic Modulation Classification Using Moments and Likelihood Maximization. IEEE Communications Letters, 22(5), 938-941. DOI: 10.1109/LCOMM.2018.2806489.
22. Shimbo, D., Oka, I., Ata, S. (2007). A modulation classification using joint moments with linear transform. In 2007 IEEE Radio and Wireless Symposium, pp. 567- 570. https://doi.org/10.1109/RWS.2007.351894

23. Li, C., Xiao, J., Xu, Q. (2011). A novel modulation classification for PSK and QAM signals in wireless communication. In IET International Conference on Communication Technology and Application (ICCTA 2011), pp.89-92.https://doi.org/10.1049/cp.2011.0636.
24. Automatic Modulation Classification Using a Support Vector Machine-Based Pattern Recognition Algorithm Varna Kumar Reddy , P.G. Meena M. Department of ECE, Vels Institute of Science, Technology & Advanced Studies, Chennai 600117, India. https://doi.org/10.18280/isi.xxxxxx
25. M. W. Aslam, Z. Zhu and A. K. Nandi, “Automatic Modulation Classification Using Combination of Genetic Programming and KNN,” IEEE Transactions on Wireless Communications, vol. 11, no. 8, pp. 2742-2750, August 2012.
26. Venkata Subbarao, M., Samundiswary, P. (2021). Automatic modulation classification using cumulants and ensemble classifiers. In Advances in VLSI, Signal Processing, Power Electronics, IoT, Communication and Embedded Systems, pp. 109-120. https://doi.org/10.1007/978-981-16-0443-0_9