Q1. To predict if it will rain tomorrow in XYZ country using suitable ML approach

The dataset contains 145460 rows and 23 columns.

We can notice that the dataset contains a lot of ‘NA’ values which affects the model training.

So our first step will be to somehow replace these ‘NA’ values rather than dropping them. Because ‘NA’ values add up to 3,43,248 rows. Dropping them can lead to loosing information from the dataset and the model may not be able to fit well.

⇒ Preprocessing the non-ordinal categorical variables into separate columns using *pd.get_dummies()* function. This separates the columns values into separate columns making them suitable for training purposes.

⇒ Preprocessing the “Date” column into separate columns as “Day”, “Month”, “Year”.

Methods for replacing NA values and training

→ Mean and mode :

numeric_features are ['MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation', 'Sunshine', 'WindGustSpeed',
'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm', 'Pressure9am',
'Pressure3pm', 'Cloud9am', 'Cloud3pm', 'Temp9am', 'Temp3pm']

categorical_features are ['WindGustDir', 'WindDir9am', 'WindDir3pm', 'RainToday','RainTomorrow']

We replace the numerical variables using mean of the column and categorical variables using mode of the column.

Models used :

We do a train - test split on the dataset with test_size =0.2 and random_state = 40.

i) Decision tree - This could be best model in terms of performance for this task., since the dataset is huge and have numerous number of independent classes.

ii) Random forest - This is a tree based bagging technique that uses ensemble learning method. So it can also be used. We can also use parameter tuning using grid search.

→ Interpolate using pandas data frame:

It uses statistical techniques to interpolate the missing values. In which we use - ‘linear’ method that ignores the index and treat the values as equally spaced. This is the only method supported on MultiIndexes.

One drawback of this technique is that it tries to fill the cells and if it is not possible it leaves it as it is as “NA”.

So, we should again replace “NA” values with mean and mode.

On the above dataset we can perform training again.

Models used :

We do a train - test split on the dataset with test_size =0.2 and random_state = 40.

i) Decision tree - This could be best model in terms of performance for this task., since the dataset is huge and have numerous number of independent classes.

ii) Random forest - This is a tree based bagging technique that uses ensemble learning method. So it can also be used.

→ Iterative Imputer

Since the missing values are large, we use **Iterative imputer** - Iterative Imputer is a multivariate imputing strategy that models a column with the missing values (target variable) as a function of other features (predictor variables) in a round-robin fashion and uses that estimate for imputation.

On the above dataset we can perform training again.

Models used :

We do a train - test split on the dataset with test_size =0.2 and random_state = 40.

i) Decision tree - This could be best model in terms of performance for this task., since the dataset is huge and have numerous number of independent classes.

ii) Random forest - This is a tree based bagging technique that uses ensemble learning method. So it can also be used.

→ ANN model training

We use a special technique called cyclic encoding of day to transform date while using for ANN that gives better results.

Further we use mean and mode for replacing NA values.

Then, we use label encoder in sklearn package to encode categorical variables with value between 0 and n_classes-1.

Then, we use standard scaler to scale the numerical variables.

Then, we remove outliers using IQR technique for each column.

We then use, Keras library to compile a neural network using Sequential( ).

Fit the model and perform evaluation metrices on test data.

→ KNN imputer

It performs Imputation for completing missing values using k-Nearest Neighbors.

We use label encoder in sklearn package to encode categorical variables with value between 0 and n_classes-1.

Then use KNN imputer and fit transform to impute “NA” values.

Since KNN imputer is prone to outlier, we use IQR to remove outliers from the dataset.

Models used:

We do a train - test split on the dataset with test_size =0.2 and random_state = 40.

i) Random forest - This is a tree based bagging technique that uses ensemble learning method. So it can also be used.

ii) SVM - Used SVM with polynomial kernel with degree 3.

iii) Logistic regression - Since logistic regression can perform binary classification. We try using it as well and compare with other models.

iv) KNN - It uses proximity of the clusters from the data point to be classified. It is also a valid technique for this task.

v) Naive Bayes - It is a probability based technique, though it doesn't work well on numerous independent variables. Lets just give it a try.

Inferences:

⇒ The correlation matrix denotes that the dependent variable "Rainfall" is not fully dependent on a single variable. All the independent variables affect the "Rainfall" almost equally.

⇒ Th following tables denote the accuracy of the various models trained under various "NA" filling techniques.

	Mean, Mode	Interpolate	Iterative imputer
Decision Tree	0.785	0.784	0.790
Random forest	0.851	0.850	0.853

ANN - 0.853

	Kernel Imputer
Random forest	0.860
SVM	0.844
Logistic regression	0.845
KNN	0.842
Naive bayes	0.784

From the above data it can be inferred that kernel imputer with Random forest model performs the best with accuracy score of 0.860 .