POST-ML Session With Deployment

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Content

- ML Preview of Classification
- ML Preview of Some ML Algorithms
- Hands on Deployment of Ensemble Model
- Review Quiz & Assignment

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

What is Supervised Learning?

- Goal: Predict target variable(label) given the predictor variables(features)
- Classification - Target variable is categorical
- Regression - Target variable is continuous
- Purpose

  - automate time consuming and expensive manual tasks - eg. a doctor's diagnosis

  - Make predictions about the future- eg. will a customer click an ad or not

- Main requirement - labelled data (historical, experimental)

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Scikit Learn for Machine Learning

- All ML models are implemented as Python Classes
- They implement the algorithms for learning and predicting
- Store information learned from the data
- Training the model on the data = fitting a model to the data –using .fit() method
- Predicting the labels of new data– using .predict() method
- Scikit Learn API requires
  - Data as a numpy array or pandas dataframe
  - Features be numerical
  - No missing values

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Algorithms for Classification

- K Nearest Neighbors
- Logistic Regression
- Decision Trees
- Random Forest
- Naive Bayes
- Gradient Boosting
- Adaboost Classifier
- XgBoost Classifier

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

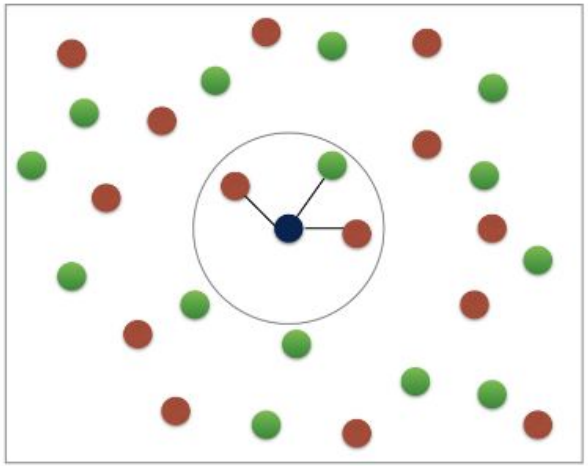
KNN

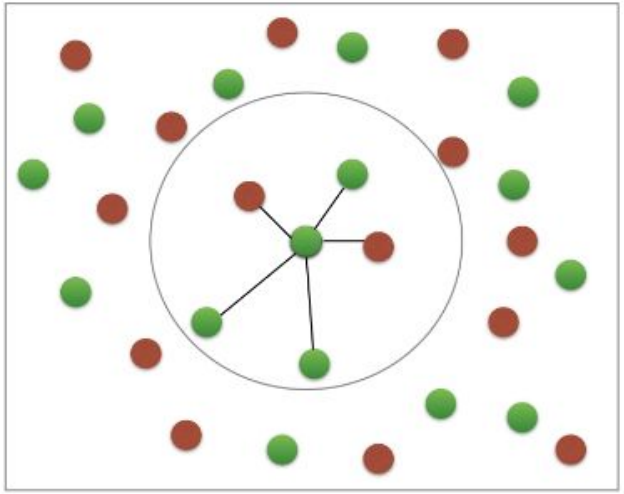
- The k-nearest neighbors (KNN) algorithm is a supervised machine learning algorithm.
- KNN assumes that similar things exist in close proximity. This implies that similar data points are close to each other. KNN calculates the distance between points on a graph to decide similarity
- The distance can be of any type, e.g., Euclidean, Manhattan, etc.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

K- Nearest Neighbors

- Basic Idea : predict the label of a data point by
  - Looking at “k” closest labelled data points
  - Taking a majority vote





Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

KNN Advantages and Disadvantages

Advantages

- Very easy to implement.
- This algorithm can be used for both classification and Regression.
- Since data is not previously assumed, it is very useful in cases of nonlinear data.
  The algorithm ensures relatively high accuracy.

Disadvantages

- It is a bit more expensive as it stores the entire training data.
- High memory storage requirements for this algorithm.
- Higher sets of values may lead to inaccurate predictions.
- Highly sensitive to the scale of the data.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

KNN Uses

The following are some of the areas in which KNN can be applied successfully:

- KNN is often used in banking systems to identify if an individual or organization is fit for a grant or a loan based on key characteristics.
- KNN can be used in Speech Recognition, Handwriting Detection, Image Recognition, and Video Recognition.
- A potential voter can be classified into categories based on characteristics (like “voter” or “non-voter”) for elections

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Naïve Bayes

What is it ? Statistical method for classification.
Supervised Learning Method.
Assumes an underlying probabilistic model, the Bayes theorem. Can solve problems involving both categorical and continuous valued attributes
Naive Bayes classifier works on the principles of conditional probability as given by the Bayes theorem

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Naïve Bayes

Bayes Theorem gives the conditional probability of an event A given another event B has occurred

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Where, P(A/B) = Conditional Probability of A given B

P(B/A) = Conditional Probability of B given A

P(A) = Probability of event A

P(B) = Probability of event B

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Example : To Play or Not to Play

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Example : To Play or Not to Play

- Look up tables

P(Outlook=*Sunny* | Play=*Yes*) = 2/9
P(Temperature=*Cool* | Play=*Yes*) = 3/9
P(Huminity=*High* | Play=*Yes*) = 3/9
P(Wind=*Strong* | Play=*Yes*) = 3/9
P(Play=*Yes*) = 9/14

P(Outlook=*Sunny* | Play=*No*) = 3/5
P(Temperature=*Cool* | Play==*No*) = 1/5
P(Huminity=*High* | Play=*No*) = 4/5
P(Wind=*Strong* | Play=*No*) = 3/5
P(Play=*No*) = 5/14

- MAP rule

$P(Yes \mid \mathbf{x}')$: [P(*Sunny* | Yes)P(*Cool* | *Yes*)P(*High* | Yes)P(*Strong* | *Yes*)]P(Play=*Yes*) = 0.0053
$P(No \mid \mathbf{x}')$: [P(*Sunny* | No) P(*Cool* | No)P(*High* | *No*)P(*Strong* | *No*)]P(Play=*No*) = 0.0206

Given the fact $P(Yes \mid \mathbf{x}') < P(No \mid \mathbf{x}')$, we label $\mathbf{x}'$ to be "*No*".

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Advantages vs Disadvantages of Naïve Bayes

Advantages

This algorithm works quickly and can save a lot of time.

Naive Bayes is suitable for solving multi-class prediction problems.

Naive Bayes is better suited for categorical input variables than numerical variables.

Disadvantages

Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. This limits the applicability of this algorithm in real-world use cases.

This algorithm faces the ‘zero-frequency problem’ where it assigns zero probability to a categorical variable whose category in the test data set wasn’t available in the training dataset. It would be best if you used a smoothing technique to overcome this issue.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Use Cases of Naïve Bayes

Text Classification

Most of the time, Naive Bayes finds uses in-text classification due to its assumption of independence and high performance in solving multi-class problems. It enjoys a high rate of success than other algorithms due to its speed and efficiency. It is also an excellent spam filter for emails.

Sentiment Analysis

One of the most prominent areas of machine learning is sentiment analysis, and this algorithm is quite useful there as well. Sentiment analysis focuses on identifying whether the customers think positively or negatively about a certain topic (product or service).

Recommender Systems

With the help of Collaborative Filtering, Naive Bayes Classifier builds a powerful recommender system to predict if a user would like a particular product (or

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Decision Trees

A supervised Algorithm which uses tree structure to model relationships among the features and the potential outcomes

It breaks down dataset into smaller subset with increase in depth of tree

It’s a flowchart for deciding how to classify a new observation

Consists of Root ,branches and leaves

Takes top down greedy approach known as recursive binary splitting

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Loss Function for Decision Tree

For classification the loss function is measured in terms of a measure of randonmess known as Entropy .

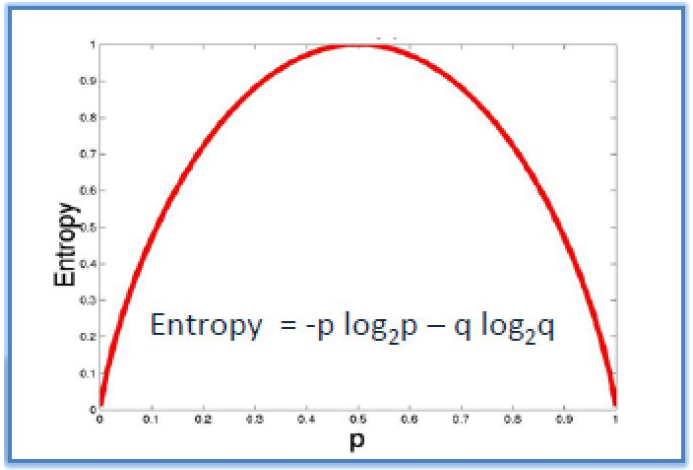
The entropy graph is shown below we see that entropy is maximum when p=0.5 and minimum when p=0 or p=1

Entropy is given by the formula

$$\sum_{i=1}^{C} - p_i \log_2 p_i$$

Where i=1, 2,3… C

If C=2 it means the feature has 2 classes



Entropy = -0.5 $\log_2$0.5 – 0.5 $\log_2$0.5 = 1

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Gini Index

A measure of impurity similar to entropy

Lower the value of gini index higher the homogeneity

Calculate gini index for sub-nodes

Calculate gini for split using weighted Gini score of each node of that split

Formula - $$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Decision Tree Algos

ID3 (Iterative Dichotomiser 3) – developed by Ross Quinlan. Creates a multi branch tree at each node using greedy algorithm. Trees grow to maximum size before pruning

C4.5 succeeded ID3 by overcoming limitation of features required to be categorical. It dynamically defines discrete attribute for numerical attributes. It converts the trained trees into a set of if-then rules. Accuracy of each rule is evaluated to determine the order in which they should be applied

CART (Classification & Regression Trees) is similar to C4.5 but it supports numerical target variables and does not compute rule sets. Creates binary tree. Scikit uses CART

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Implementation Steps

Calculate entropy for each branch or split

Split the dataset by selecting attribute with highest information gain

Repeat the above steps on each branch

A branch with entropy of zero is leaf node

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Pruning

Decision tree tends to get overfit with training sample and becomes too large and complex. A complex and large tree poorly generalizes the new samples data whereas a small tree fails to capture the information of training sample data.

Pruning may be defined as shortening the branches of tree. The process of reducing the size of the tree by turning some branch node into leaf node and removing the leaf node under the original branch.

Pruning is done by controlling the hyperparameter called max depth

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Hyperparameters of Decision Trees

Maximum Depth- the largest length between the root to leaf. A tree of maximum depth k can have at most 2**k leaves.

Minimum number of samples per leaf- we can set a minimum for the number of samples we allow on each leaf.

Minimum sample split - the minimum number of samples required to split an internal node

Criterion – ‘gini’ or ‘entropy’

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Pros and Cons of Decision Trees

PROS

Easy to explain especially to non technical people
Highly interpretable as it is a visual model
Can deal with Categorical and Numerical Outcome Variables
It is fast, efficient and a low bias model

CONS

Easily influenced by outliers
High Variance model and thus very prone to overfitting
Low Predictive accuracy

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Random Forest

Random forests are a form of ensemble learning where we use the concept of bagged trees

Random forests provide an improvement over bagged trees by way of a random small tweak that decorrelates the trees.

As in bagging, we build a number of decision trees on bootstrapped training samples

But when building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors. A fresh sample of m predictors is taken at each split, and typically we choose $m \approx \sqrt{p}$

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

PROS of Random Forest

It reduces variance as taking many trees from n bootstrapped samples and averaging their prediction leads to lower variance

It also reduces bias in the dataset by decorrelating the trees which is a problem with imbalanced datasets

Random forests overcome this problem by forcing each split to consider only a subset of the predictors. Therefore, on average (p − m)/p of the splits will not even consider the strong predictor, and so other predictors will have more of a chance. We can think of this process as decorrelating the trees, thereby making the average of the resulting trees less variable and hence more reliable.

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Hyperparameters for Random Forest

Criterion- can choose between ‘entropy and ‘gini’

Maximum Depth- the largest length between the root to leaf.

Minimum number of samples per leaf- we can set a minimum for the number of samples we allow on each leaf.

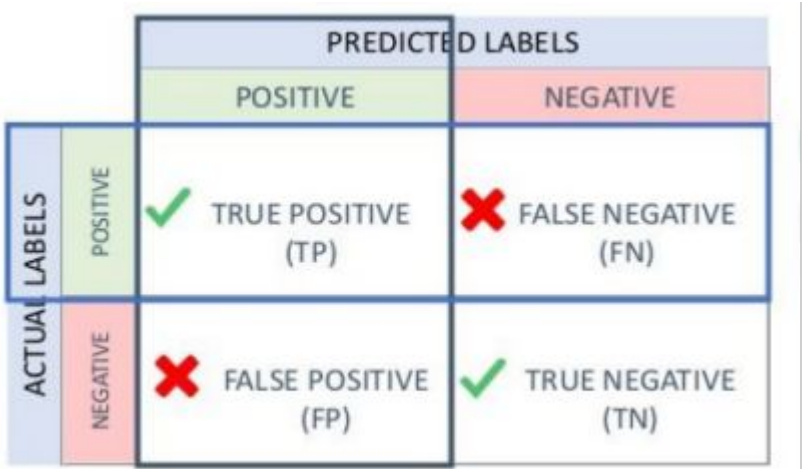
Minimum sample split - the minimum number of samples required to split an internal node

Maximum features - the number of features that one looks for in each split. We can shoose between ‘sqrt’ and ‘log2’

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Evaluation Metrics for Classification- Confusion Matrix

- Precision and Recall are metrics that are used for evaluating the performance of a classification algorithm. These can be understood more clearly using a confusion matrix as shown beside.
- Precision is the proportion of true positives among those predicted positives. So Precision is a % expressing the accuracy with which positive classes are predicted. As a formula Precision = TP/(TP+FP)
- Recall on the other hand is the proportion of true positives among those that are actually positive. So recall is a % expressing the capacity of the model to recall positive values. As a formula Recall =TP/(TP+FN)
- These measures are very useful in a marketing scenario



Hands on Deployment

We will create a medical diagnostic web app by working on the Pima Diabetes Dataset and create a classification model and will deploy it using streamlit

Dependencies

1. Jupyter Notebook
2. VS Code
3. Github account local Git Bash
4. Streamlit sharing account connected to Github (share.streamlit.io)

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Assignment

Create an optimized model to predict employee attrition and deploy it as a streamlit model

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.

Reference Books

1. Introduction to Statistical Learning by Gareth James (Springer Publications)
2. Introduction to Machine Learning with Python by Andreas Muller & Sarah Guido (O'Reilly Publication)

Proprietary content. © Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.