**Data Mining**

**CONTEXT**

* **EDA**
* **DATA PREPARATION**
* **PREDICTIVE ANALYSIS**
* **EVALUATION OF PREDICTIVE ANALYSIS**
* **FEATURE SELECTION**
* **DECRIPTIVE ANALYSIS**
* **RECOMMENDER SYSTEMS**
* **TEXT MINING**
* **INTRODUCTION TO DEEP LEARNING**
* **VISUAL ANALYTICS\**

**Data And information**

* **Data – Recorded facts and figures**
* **Information – Set of patterns , or expectection , that underlie**

**Data on its own carries no meaning , It is tangible while information is intangible It is taking out meaning from data**

**Growing gap between the generation of data and our understanding of it**

**Motivatation for data mining**

* **Looking for patterns in data**
* **Intelligently analysed data is a valuable resource**

**Data Mining**

* **The Extraction of Implicit , Previously unknown , potentially useful information from data**

**Data Mining Workflow or Cross industry Standard or CrispDm**

1. **Gathering of data and Business Understanding and data Understanding**
2. **Data preparation for machine learning**
3. **Modelling with algorithms – predictive , descriptive , prescriptive**
4. **Evaluation**
5. **Deployment**

**Diagram

Description automatically generated**

**Data Mining Vs Machine Learning**

1. **Data mining**

**At its most complex level , data mining involves machine learning which can be used for descriptive and predictive analysis**

**- Revealing Hidden patterns in data (Descriptive models)**

**- Making Prediction (Predictive models)**

1. **Machine Learning**

**Data Mining i/0**

* **Input :Typically, single table with data**

**Data rows represent independent from each other**

**Colums measure aspects of an instanct**

* **Output : Compact structural patterns that describes the concept**

**Can be used for automatic decision-making to predict the value of a particular attribute of new instances**

**\* There should be no relation between data rows**

**Types of Attributes**

* **Categorical attribute :values serve only as labels or names**

**- Nominal**

**- no relation is implied among values**

**- Oridnal**

**- Ordering among values is implied**

* **Numerical attributes : Discrete or continuous**

**Data Mining Challenges**

* **There should be sufficient amount of data for discovery of useful information**
* **Large dataset may require management: distributed storage and processing**
* **Discovered patterns in data can e**

**- Trivial and uninteresting**

**- untrue, dependent on accidental coincidences in the particular dataset**

* **Real data is imperfect**

**\* Therefore Train and test split is used for not being biased and remove overfit issues**

**\* The data mining process needs to be robust enough to cope with imperfect data**

**Exploratory Data Analysis**

**At a high level, EDA is the practice of using visual and quantitative methods to understand and summarize a dataset without making any assumptions about its content.**

**Best practice**

* **Explore the data – missing data, outliers, noise.**
* **Understand the relationship , or lack of , between columns**
* **Identify useless columns**

**\* do not transform the dataset in EDA Phase**

**\* Make conclusion but not on based of ignoring part of data**

**Visit :-** [**https://github.com/sarthakpunjabi/data\_science\_projects/tree/main/EDA\_Project**](https://github.com/sarthakpunjabi/data_science_projects/tree/main/EDA_Project)

**Practical Observation**

* **Median – even( n/2 + n+1/2) odd(n/2)**
* **Sns.boxplot(data,orient=’v’, width)**
* **Q1 – 1.5\*3(Q3-Q1) Outliers for below**
* **Q3 + 1.5\*3(Q3-Q1) Outliers for above**
* **Violinplots are for showing the distribution density**

**Pivot Table – More of a transformation of data table and applying aggregate functions on it to see a clear number and picture in a tabular form**

* df**.**pivot\_table(values**=**'Loan\_Status',
* index**=**'Credit\_History',
* aggfunc**=lambda** x: x**.**map({'Y':1, 'N':0})**.**mean())

Data Preparation / Data Wrangling / Data munging

* Treating missing values
* Treat outliers
* Feature engineering

- transform exitsting feature

- Adding new feature

Types of missing values

* MCAR – Missing Completely At Random
  + missing values are randomly distributed across all observations
* MAR – Missing At Random
  + data is missing randomly only within sub-samples of data
* MNAR – Missing Not At Random
  + missing data has a structure to it

Treatment of missing values

* Impute missing values
  + Simple method: use mean, median, most frequent, constant
  + Better method: KNN (K-Nearest Neighbour)

Handling Outliers

* Drop rows with outliers
  + Cons: information loss
* Add a new binary feature to mark outliers
* Transform
  + Log transformation
  + Cubic root transformation

For Cyclical Data

* Use a sine and cosine transformations
* Example: <https://towardsdatascience.com/how-to-handle-cyclical-data-in-machine-learning-3e0336f7f97c>
* import numpy as np last\_week['Sin\_Hour'] = np.sin(2 \* np.pi \* last\_week['Hour'] / max(last\_week['Hour'])) last\_week['Cos\_Hour'] = np.cos(2 \* np.pi \* last\_week['Hour'] / max(last\_week['Hour']))

Rescale Features and Normalization

* Many ML algorithms assume features are on the same scale.
* Scikit-learn Scalers:
  + MinMaxScaler
  + RobustScaler
  + StandardScaler
* Normalisation
  + Normalise data rows (when many features have the same nature, e.g. word frequency in text mining)

Min Max Scaller

* Z = (x-min)/(max-min)
* Range typically (0-1)

Robust Scaler

* Z = (x – median)/(IQR) = (x-q2)/(q3-q1)
* It doesn’t squeeze to 0-1
* No force
* Use RobustScaler if you want to reduce the effects of outliers, relative to MinMaxScaler.

Standard Scaler

* Z = (x-u)/s

\*\* Treating Outliers using Log transformation

Normalizer

* Applied to rows not columns
* Each row of the dataset with at least one non zero component is rescaled independently of other rows so that its norm(L1,L2 or Linf)

- L1 is sum of all absolute values in the rows

- L2 is the square root of sum of all square values in the rows

- Linf also known as max norm is equal to the maximum absolute value in the row

Style of Machine Learning

* Predictive/Supervised
* Descriptive/Unsupervised
* Semi supervised
* Reinforcement Learning

1. Predictive

* Classification Techniques
* Numeric prediction techniques

1. Descriptive/Unsupervised

* Detecting associations between features(Association Learning)
* Clustering Techniques

Clustering : K-Means

Aim – Divide instances into “ natural” groups

* Clusters can be disjoint vs overlapping
* Deterministic vs probabilistic
* Flat vs hierarchial

Definition

* K means minimizes the sum of squared distances from the data points to the centroids . The sum is called inertia
* To increace the chance of finidng a global minimum of he inertia

1. Restart with different random centers
2. Smart choice of initial Kmeans++

Properties of meaningful k means clustering

* Centroid should be as far as possible
* Interia = sum of squared distances from the cenroid to data points
* Dunn like indices = min(Inter cluster distance) / Max(Inter cluster distance)(Should be maximized)
* Silhouette coefficient = s = b – a / max(a,b)

Best Value of K

* The values of these metrics (intertia , Dunn index ,silhouette coefficient)
* Inertia should be minimum while dunn and silhouette must me max
* Split one cluster into two sub clusters at each bisecting step until a meaningful set of clusters is obtainer

Correlation

Person method = (x – mean ) \* (y – ymean) / (x – xmean)2 \* (y – ymean)2

Classification and Regression

Graphical user interface

Description automatically generated

* Predictors
* Class attribute or dependent variable

Some of the Classification

* OneR – One rule algorithm
* Naïve Bayes – All attribute contribute equally and independently
* Decisions Tree classifiers

Decision tree –

Criterion for attribute selection

1. Which is the best attribute
2. Want to get the smallest tree
3. Heuristic:Choos the attribute that produces the purest nodes
4. Measure of purity

* Info[node] – information value of a node measured in bits
* The amount of further information necessary to take a decision at tree\_node

1. Strategy : Choose attribute with the lowest information value of its child nodes

How to measure information value

* Entropy function is used to measure information value
* When impurity is maximal that is all are classes equally likely measure should be maximal
* (p1,p2….n) = -p1logp1 – p2logp2
* Eg – (4/4,0/4) = -(1 \*log1 + 0 \* log 0 )
* Eg - (2/5,3//5) = -(2/5) \* log(2/5) –(3/5)\* log(3/5)

Graphical user interface, text

Description automatically generated

* Information Gain. = Before Gain – After Gain

This is Top Down induction of decision tree : ID3 algorithm

We should use pruning technique to avoid overfitting

There is another similar approach called as Cart

\* this is for categorical data

\* for Numerical data there is C4.5

Random Forest

* Ensemble machine learning algorithm called bootstrap aggregation or bagging
* It uses bagging and feature randomness when building each individual tree to create uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree

Ensemble learning – It referes to a group of base ML algorithm which work collectively to achieve better predictive model

There are two main types

1. Bagging – the models are trained in parallel
2. Boosting – The base models are trained sequentially

KNN – Instance Based Learning/rote/lazy learning

Training , validation and Testing

* Typically learning schemes operate in multiple stages
* Split the data into training set, validation set , ad test set
* For Each classifier : Train multiple models with different parameters on the training set and evaluate on the validation set
* For Each classifier

1. Put the training and validation sets together train the model with the best set of parameters
2. Evaluate the models on the test set to pick the best classifier have final evaluation results.

\*\* It is important that data used for training is not used in in any way to evaluate the model

Some terminology

* Natural performance measure for classification models :error rate – proportion or errors made over the whole dataset
* Resubstituion error: error rate obtainer from training data

Making the most of the data

Generally the larger the training data better the classifier

Larger the test data more accurate the error estimate

Insufficient training data increases the risk of

* Overfitting
* Underfitting

Insufficient test data results in low confidence in the result

Holdout estimation – Reserves a certain amount for testing/validation and uses the remainder for training

Usually 1/3 for testing and rest for training

Stratification – Ensures each class is represented with equal data

Repeated holdout – There are multiple iteration a certain proportion is randomly selected for training (possible for stratification)

The error rate on the different iterations are averaged to yield an overall error rate

Drawback – Overlapping

Cross Validation

When you have training set Split it into K subset you train K times K model

Steps

1. Split data into K subsets of equal size
2. Use each subset in turn for validation , the remainder for training

Subsets are stratified before cross validation

Standard method stratified ten fold cross-validation

Why ten – there is some theoretical evidence

Stratification reduces the estimate’s variance

Accuracy is not all

* When incurs cost

Confustion matrix

* Precision = TP/TP+FP
* TRUE POSITIVE RATE (SENSITIVITY OR RECALL) = TP/TP+FN
* SPECIFICITY = TN/(TN+FP)
* F1 SCORE = (2 \* RECALL \* PRECISION )/ (PRECISION +RECALL)

COST SENSITIVE CLASSIFICATION

Two Example cost matrices

Imbalance distribution – precision recall curve

Normal distribution – ROC curve

Linear regression

* There should be a continuous
* K+1 predictors
* X. = w0 + w1a1 + w2a2 +wkak (Linear combination where w1 w2 is coefficient)
* Weights are calculated from the training data
* Predicted value for the first training instance

Minimizing the square error

Mean square error = (p1 – a1)2 + ….. + (pn – an)2 /n

Root mean square error

Mean absolute error

Relative error

Correlation coefficient