FREE Online
Citizen Data Scientist Course

You Too Can be a Citizen Data Scientist – No Matter Your Role, Skill or Job Function

Section 8 - Analytical Techniques - Part 2

Instructor Notes: This supporting documentation includes a complete set of the slides used in the course material. Some sections also include expanded material, articles and documentation to further student understanding of more complex topics.

For your conenience, contact information is displayed at the end of the online course and at the conclusion of the supporting documentation in Section 12. We invite you to contact us with questions, requests or comments.

Section 8 - Analytical Algorithms and Techniques – Part 2

Forecasting

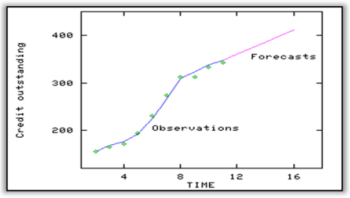
Algorithm Techniques

Forecasting

Forecast values for the future based on past values, with one or more variables affecting future values.

Algorithms:

- Holt-Winters Exponential Smoothing
- ARIMA
- ARIMAX



Example: Forecast product sales based on past sales, inflation, and GDP growth.

Forecasting – Holt Winters

Introduction

$$0.9^1, 0.9^2, 0.9^3, 0.9^4, 0.9^5, 0.9^6 \ldots$$
$$\text{or: } 0.9, 0.81, 0.729, 0.6561, 0.59049, 0.531441, \ldots$$

- The idea of Holt Winters exponential smoothing is to smooth the original univariate series and to use the smoothed series in forecasting future values of the variable of interest.
- Exponential Smoothing assigns exponentially decreasing weights as the observation get older. In other words, recent observations are given relatively more weight in forecasting than the older observations.
- For example, Imagine a weighted average where we consider all of the data points, while assigning exponentially smaller weights as we go back in time. For example if we started with 0.9, our weights would be (going back in time):

Forecasting – Holt Winters

Introduction

Holt Winters Algorithms are applied when data is stationary.

Basically, there are three types of Holt Winters Exponential smoothing methods :

1. Holt Winters Single Exponential Smoothing : Suitable for forecasting data with no trend or seasonal pattern. However level of the data may be changing over time as shown in figure 1
2. Holt Winters Double Exponential smoothing : suitable for Forecasting data with trend as shown in figure 2.
3. Holt winters Triple Exponential Smoothing : Suitable for forecasting data with trend and/or seasonality as shown in figure 2.

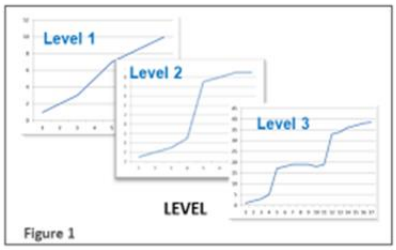


Figure 1

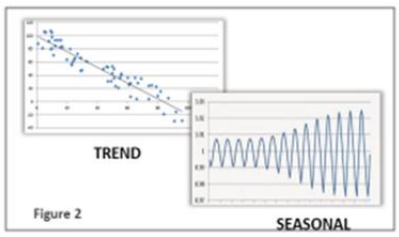


Figure 2

Forecasting – Holt Winters

Use Case

- Business problem :
- Forecasting number of viewers by day for a particular game show for next two months
- Input data : Last six months daily viewer count data
- Data pattern : Data taken as input exhibit stationarity and no trend/seasonality
- Business benefit :
- Help in planning for repeat telecast
- Can pitch in for more advertisement (fund raise) if projected count of viewers are high
- Improvement planning can be done for the game show to increase/maintain the level of popularity

Forecasting – Holt Winters

Limitations

- Holt Winters Exponential smoothing Algorithms are used only when data series is stationary
- Holt Winters Single exponential Smoothing is used for forecasting data with only levels and in real life data this is hardly the case
- Holt Winters is only for univariate data forecasts

Forecasting – ARIMA

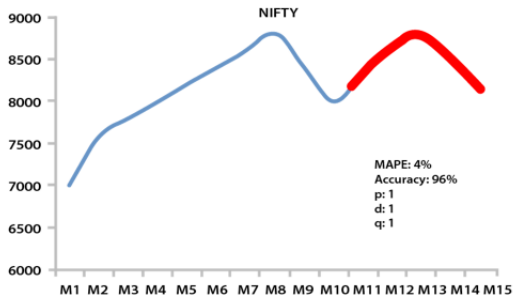
Introduction

- ARIMA stands for Autoregressive Integrated moving Average model
- An **A**uto **R**egressive **I**ntegrated **M**oving **A**verage (ARIMA) model predicts future values of a time series by a linear combinations of its past value and a series of errors.
- This method is suitable for forecasting when data is stationary/non stationary, univariate and has any type of data pattern : **level/trend/seasonality/cyclicity**

Forecasting – ARIMA

Sample UI for output

Output will contain forecasted values based on user specified time period along with line chart showing actual and forecasted series, prediction accuracy and parameters values decided by the algorithm

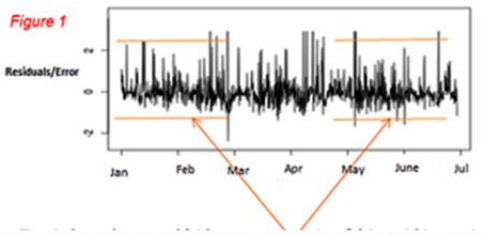


Actual values	
Months	Index
M1	7000
M2	7600
M3	7800
M4	8000
M5	8200
M6	8400
M7	8600
M8	8800
M9	8400
M10	8000
Forecasted values	
M11	8350
M12	8650
M13	8780
M14	8480
M15	8160

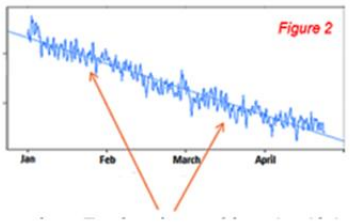
Forecasting – ARIMA

Limitations

- ARIMA is only for univariate data forecasts, but there might be other variables affecting the output/dependent variable. ARIMA doesn't take into account the influence of other predictors while forecasting, hence forecasting made might not be accurate
- There should be at least 50 observations in the input data
- ARIMA model is suggested for short term forecasting
- The Forecast error also known as "Residuals" should show nearly constant trend over time i.e. it should be time independent as shown in the figure 1 below in contrast to the increasing/decreasing trend shown in figure 2 below:



Time independent error (fairly constant over time & lying within certain range)



Time dependent error (decreasing with time)

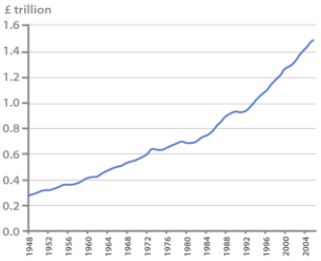
Forecasting – ARIMA

Example

Let's take an example of year wise GDP values of India.

As shown in figure below, the plot of these data suggests that this is non stationary data with upward trend.

Hence, we can choose ARIMAX algorithm for forecasting GDP as there would be more than one variable affecting the GDP



Actual GDP (Trillion)	
Years	GDP
Y1	0.35
Y2	0.38
Y3	0.39
Y4	0.40
Y5	0.44
Y6	0.50
Y7	0.58
Y8	0.60
Y9	0.64
Y10	0.70
Forecasted GDP (Trillion)	
Y11	0.82
Y12	0.94
Y13	1.00
Y14	1.22
Y15	1.42

Forecasting – ARIMA

Introduction

- An Autoregressive Integrated Moving Average with Explanatory Variable (ARIMAX) model can be viewed as a multiple regression model with one or more autoregressive (AR) terms and/or one or more moving average (MA) terms
- This method is suitable for forecasting when data is stationary/non stationary, Multivariate and has any type of data pattern : **level/trend/seasonality/cyclicity**
- ARIMAX is simply an ARIMA with additional explanatory variables in categorical and/or numeric format

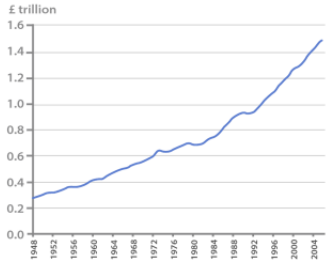
Forecasting – ARIMA

Example

Let's take an example of year wise GDP values of India.

As shown in figure below, the plot of these data suggests that this is non stationary data with upward trend.

Hence, we can choose ARIMAX algorithm for forecasting GDP as there would be more than one variable affecting the GDP



Actual GDP (Trillion)	
Years	GDP
Y1	0.35
Y2	0.38
Y3	0.39
Y4	0.40
Y5	0.44
Y6	0.50
Y7	0.58
Y8	0.60
Y9	0.64
Y10	0.70
Forecasted GDP (Trillion)	
Y11	0.82
Y12	0.94
Y13	1.00
Y14	1.22
Y15	1.42

Forecasting – ARIMA

Limitations

It is based on an assumption of linear relationship between the predictors (X) and the target variable (Y) i.e. the scatter plot of each predictor versus target variable should be nearly as shown in the figures 1 & 2 in right

Furthermore, there should not be multicollinearity in data

- Multicollinearity generally occurs when there are high correlations between two or more predictor variables
- Examples of correlated predictor variables (also called multicollinear predictors) are: a person's height and weight, age and sales price of a car, or years of education and annual income
- An easy way to detect multicollinearity is to calculate correlation coefficients for all pairs of predictor variables, if it is close to or exactly 1 then one of the predictors should be removed from the model if at all possible

Figure 1

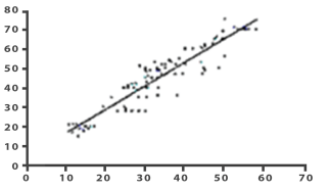
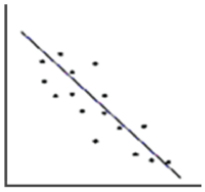


Figure 2



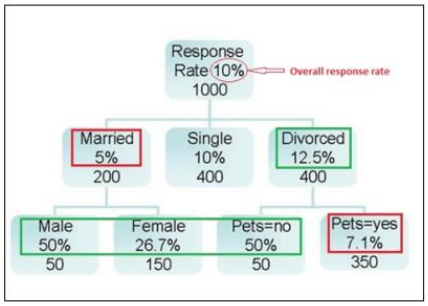
Note : Refer calculations section to understand Multicollinearity & Autocorrelation

Classification

Algorithm Techniques

Classification

Split data into groups based on pre-assigned categories or classes.



Algorithms:

- Logistic Regression
- Decision Tree
- K-Nearest Neighbour
- Naive Bayes
- Support Vector Machine

Example: An applicant for a new loan can be assigned likely/unlikely defaulter categories based on the preassigned defaulter/non-defaulter category for older applicants.

Classification – Logistic Regression

Introduction

- Objective :
  - Logistic regression measures the relationship between the categorical target variable and one or more independent variables
  - It deals with situations in which the outcome for a target variable can have only two possible types
  - Thus, logistic regression makes use of one or more predictor variables that may be either continuous or categorical to predict the target variable classes

- Benefit :
  - Logistic regression model output helps identify important factors (Xi) impacting the target variable (Y) and also the nature of relationship between each of these factors and dependent variable

Classification – Binary Logistic Regression

Example : Binary Logistic Regression : Output

Actual Versus Predicted

Classification Accuracy: (35+ 70) / (35+70+4+4) = 92%
- The prediction accuracy is useful criterion for assessing the model performance
- Model with prediction accuracy >= 70% is useful

Classification Error: 100- Accuracy = 8%
There is 8% chance of error in classification

		Predicted	
		Defaulted	Not defaulted
Actual	Defaulted	35	4
	Not defaulted	4	70

Classification – Binary Logistic Regression

Example : Binary Logistic Regression : Input

Let's conduct the Binary Logistic Regression analysis on following variables :

Target Variable (Y) | Independent Variable (X)

Default Status	Age	Marital Status	Existing Loan Staus	Income
Defaulted	58	Married	No	46,399
Not Defaulted	44	Single	No	47,971
Defaulted	33	Married	Yes	52,618
Defaulted	47	Married	No	28,717
Not Defaulted	33	Single	No	41,216
Defaulted	35	Married	No	34,372
Not Defaulted	28	Single	Yes	64,811
Not Defaulted	42	Divorced	No	53,000
Defaulted	58	Married	No	41,375
Not Defaulted	43	Single	No	53,778
Not Defaulted	41	Divorced	No	44,440
Not Defaulted	29	Single	No	51,026

Classification – Binary Logistic Regression

Limitations

It is applicable only when target variable is categorical

Sample size must be at least 1000 in order to get reliable predictions

Binary logistic regression is not suitable when number of classes > 2

Level 1 of the target variable should represent the desired outcome.
i.e. if desired class is yes in response/non response target variable then **Yes** has to be recoded into **1** and **No** into **0**

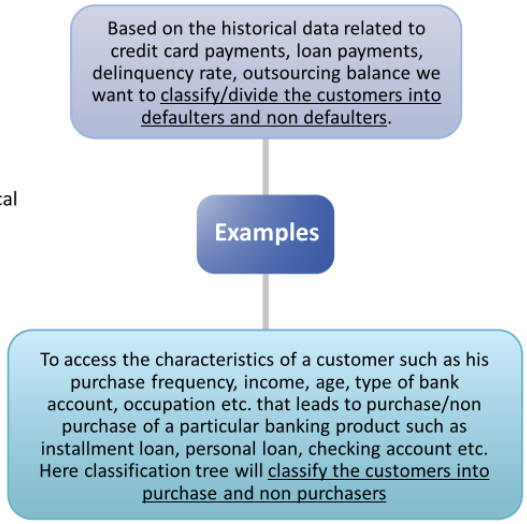
Decision Tree

Types of Decision Tree

There are two basic types of decision tree:

- Classification
- Regression

Classifications trees are needed when target variable is categorical and as the name implies are used to classify/divide the data into these predefined categories of a target variable



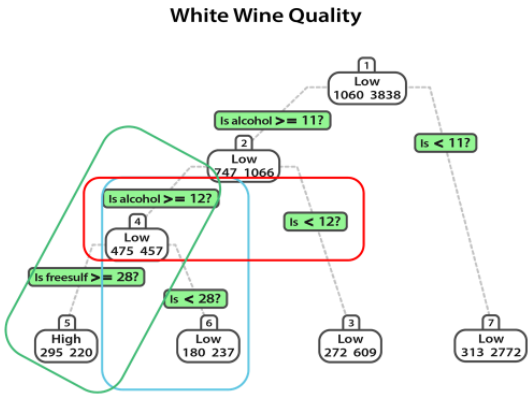
Classification Tree

How To Interpret The Classification Tree Output

Further, if alcohol >=12 then it classifies the wine to be of high quality else low quality (As seen in red box in image below)

The cases/records falling in high quality are further tested with free sulfur dioxide level. If free sulfur dioxide is >=28 and alcohol is also >=12 then such wines are classified to be of High quality. (As seen in green box in image below)

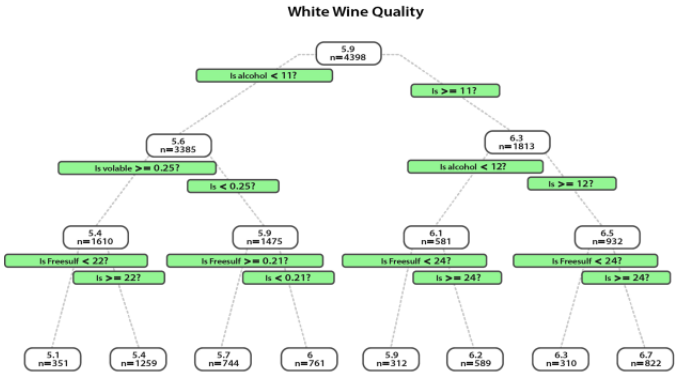
But wines with alcohol >=12 but having free sulfur dioxide >=28 are classified to be of low quality. (As seen in blue box in image below)



Classification – Regression

Method : Regression

Let's take an example of predicting the wine quality on a scale of 3 to 10 based on predictors such as alcohol level, free sulfur dioxide level, volatility etc.



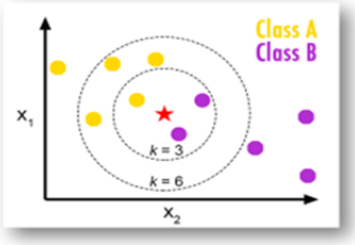
Decision Tree Classification

Limitations Of Decision Tree

Classification – K Nearest Neighbor

Introduction

- An instance (data record or case) is assigned a class, which is most common among its k nearest neighbors
- Here, k is a positive integer, typically an odd number and ranging between 1 to 10



For instance, for k = 3, the majority class of 3 nearest neighbors of center point shown as star in image is Class B (two out of three circles are purple, i.e. class B) whereas for k=6, majority is Class A (four out of 6 circles are yellow, i.e. class A)

Classification – K Nearest Neighbor

Example : Steps

Select majority class of k nearest neighbors as predicted class

Step 4,5 : As the majority class = Good for the three nearest neighbors (two out of three records have class = Good), predicted class of an instance = Good, i.e. quality of a paper tissue having acid durability = 3 and strenght = 7 is good

Final output:

Acid durability (In Seconds)	Strength (Kg/Square meter)	Paper tissue Quality
7	7	Good
7	4	Good
3	4	Bad
1	4	Good
3	7	Good

Classification – K Nearest Neighbor

Limitations :

- Data needs to be scaled [(x-min(x)/max(x)-min(x)] before inputting in the algorithm, else it can lead to high % of misclassification and in turn low accuracy
- Not suitable for classifying categorical variables
- Individual variable importance can not be measured (which variable(s) is most important or has high contribution in the classification model)
    - For instance, Age/income might be impactful variables or say, determinant factors when classifying the applicants into likely defaulters/non defaulters

Classification – Naïve Bayes

Introduction

- Naïve Bayes is a classification algorithm suitable for binary and multiclass classification
- It's a supervised classification technique used to classify future objects by assigning class labels to instances/records using conditional probability
- In supervised classification, training data already labeled with a class.
    - For example, if fraudulent transactions are already flagged in transactional data and if we want to classify transactions into fraudulent/non fraudulent then such classification is called supervised.

Classification – Naïve Bayes

How it works! - Example

- For example, a fruit may be considered to be an apple if it is red, round, and about 3" in diameter.
- Let's say we have data on 1000 pieces of fruit. The fruit being a Banana, Orange or some Other fruit and imagine we know 3 features of each fruit, whether it's long or not, sweet or not and yellow or not, as displayed in the table below:

Fruit	Long	Sweet	Yellow	Total
Banana	400	350	450	500
Orange	0	150	300	300
Others	100	150	50	200
Total	**500**	**650**	**800**	**1000**

- So from the table above we already know:
  - 50% of the fruits are bananas
  - 30% are oranges
  - 20% are other fruits

Classification – Naïve Bayes

Limitations

- Naïve Bayes classifier assumes that every features/predictor is independent, which isn't always the case
- Training dataset should be adequate enough to represent the entire population – containing every combination of class label and attributes
  - If you don't have occurrences of a class label and a certain attribute value together in training dataset (e.g. class="nice", shape="sphere") then the frequency-based probability estimate will be zero for that combination in future data
  - This problem happens when we are drawing training sample from a population and the drawn sample is not fully representative of the population
- It performs well in case of categorical input variables compared to numerical variables. For numerical variables, normal distribution is assumed which is a strong assumption.

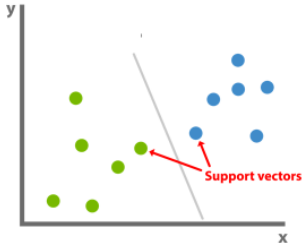
A normal distribution is an arrangement of a data set in which most values cluster in the middle of the range and the rest taper off symmetrically towards either extreme. It will look like a bell curve as shown right

Classification – SVM

Introduction

SVMs are based on the idea of finding a hyperplane that best divides a dataset into predefined classes, as shown in the image below.

The goal is to choose a hyperplane with the greatest possible margin between the hyperplane and any point within the training set, giving a greater chance of new data being classified correctly



Classification – SVM

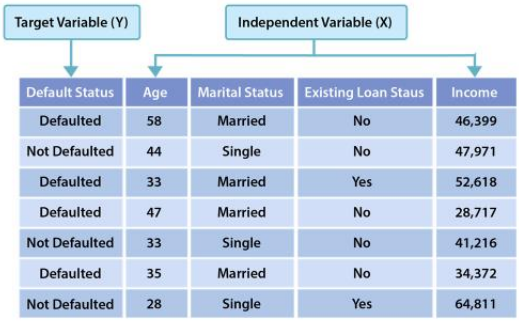
Limitations

- Processing time of SVM algorithm on large datasets can be high
- Less effective on datasets with overlapping classes

Classification – SVM

Example : Input

Let's conduct the SVM classification on following variables :



Default Status	Age	Marital Status	Existing Loan Staus	Income
Defaulted	58	Married	No	46,399
Not Defaulted	44	Single	No	47,971
Defaulted	33	Married	Yes	52,618
Defaulted	47	Married	No	28,717
Not Defaulted	33	Single	No	41,216
Defaulted	35	Married	No	34,372
Not Defaulted	28	Single	Yes	64,811