George Mason University

GBUS 738 – Data Mining for Business Analytics

Prof. Jamie Wheeler

Homework #1

Team Bench Warmers

(Gabriel Garcia, Man Lu, Jack Mchugh, Tejasvi Navnage, Abhishek Shambhu, Harshad Vatsa)

1. **Machine Learning Techniques.**

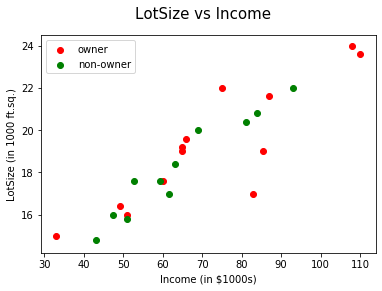
In each of the following cases, identify whether the task required is supervised or unsupervised learning, and then identify the appropriate technique—i.e., prediction, classification, affinity or clustering analysis—that you would use. Assume that an appropriate dataset is available for your algorithm to learn from.

1. Deciding whether to issue a loan to an applicant based on demographic and financial data using a database of similar data on prior customers.
   * **Supervised Learning using classification**
2. In an online bookstore, making recommendations to customers concerning additional items to buy based on the buying patterns in prior transactions.
   * **Supervised Learning using clustering**
3. Identifying a network data packet as dangerous (e.g., virus, hacker attack) based on comparison to other packets whose threat status is known.
   * **Unsupervised Learning using classification**
4. Identifying segments of similar customers.
   * **Unsupervised Learning using clustering**
5. Predicting whether a company will go bankrupt based on comparing its financial data to those of similar bankrupt and nonbankrupt firms.
   * **Supervised Learning using prediction**
6. Estimating the repair time required for an aircraft based on a trouble ticket.
   * **Unsupervised Learning using prediction**
7. Printing custom discount coupons at the conclusion of a grocery store checkout based on what you just bought and what others have bought previously.
   * **Supervised Learning using classification**

2. **Data Exploration using plots**

A company that manufactures riding mowers wants to identify the best sales prospects for an intensive sales campaign. In particular, the manufacturer is interested in classifying households as prospective owners or nonowners on the basis of Income (in $1000s) and Lot Size (in 1000 ft2 ). A marketing expert at the company undertook a random sample of 24 households (12 owners and 12 nonowners) in the city. The data are in the file RidingMowers2.xls.

a. Using Excel, Python, or R, create a scatterplot of Lot Size vs. Income, color coded by the outcome variable ownership. Make sure the plot is well formatted with legible labels, a legend, and without unnecessary whitespaces (i.e., name and format the axes appropriately).



b. Based on the available data (visually from the scatterplot), how would you classify a new household with $70,000 income and lot size 18,000 ft^2? Explain your choice.

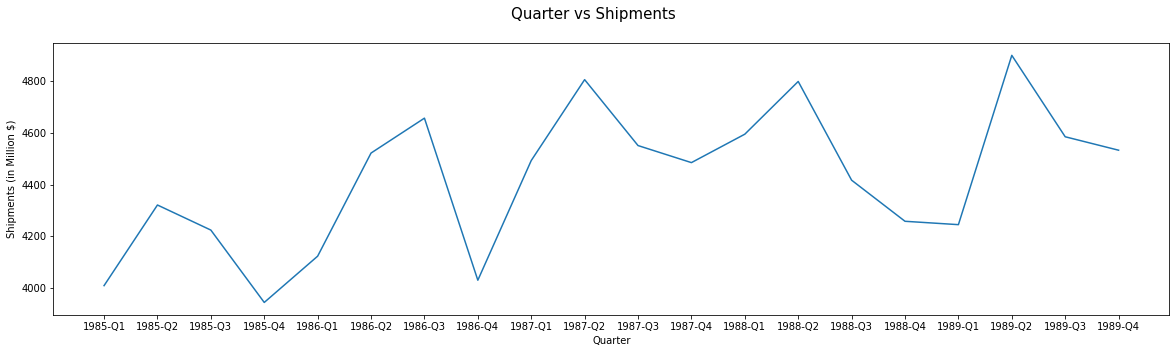
🡪We would classify a new household with $70,000 income and lot size 18,000 ft^2 as a prospective owner just by looking at the scatterplot above. Also, if we try to plot a linear line through the points it can be seen that the point is closer to the red dots (in our case owners).

However, since both the trend lines if plotted might seem to be a fit it is best to apply different machine learning algorithms which can help predict on new data based on the model.

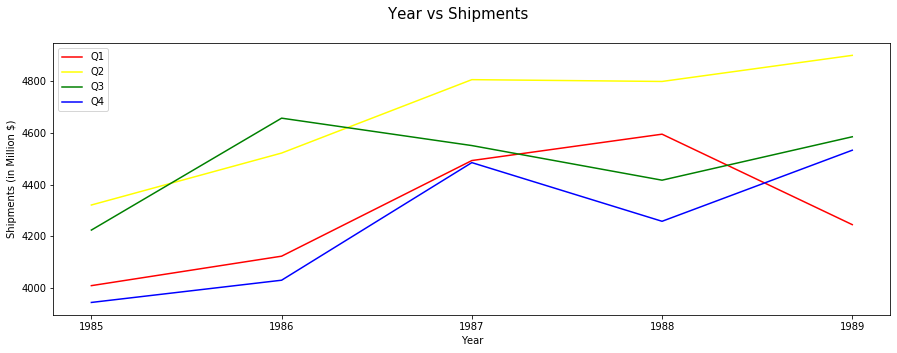
**3. Data Visualization using Excel, Python, or R**

The file ApplianceShipments.xls contains the series of quarterly shipments (in million $) of U.S. household appliances between 1985 and 1989. Use Excel, Python, or R to draw the graphs for the following problems.

a. Create a well-formatted time plot of the data. (Hint: Draw a line graph with the quarters on the X-axis and Shipments on the Y-axis).



b. Create another plot with four separate lines for the four quarters. (Hint: Sort the data by quarter and plot them as separate series.)



c. From the two graphs above, does there appear to be a quarterly pattern? Based on the pattern, what do you think the appliances might be?

🡪 Based on the two graphs above, quarters 2 and 3 tend to be higher than 1 and 4 during the period of report. Quarters 1 and 4 tend to be the colder months of the year, therefore we can predict that the appliances could be medium to big size items like dishwashers, refrigerators, washing machines and air conditioners. These items are heavier weight and require more movement and logistics planning, something that is easier when the cold temperature is not a factor.

**Reference Code:**

**﻿** #importing libraries

import matplotlib.pyplot as plt

import pandas as pd

#Q2(a)

#Importing and reading excel file to dataframe

df\_RidingMowers = pd.read\_excel("/Users/ag30103/Desktop/GBUS738/RidingMowers2.xls", delimiter=',', header=0, nrows=25, skiprows=0)

#Grouping by ownership for owner and non-owner

df\_group=df\_RidingMowers.groupby("Ownership")

group1=df\_group.get\_group("owner")

group2=df\_group.get\_group("non-owner")

#plotting scatterplot

fig = plt.figure()

fig.suptitle('LotSize vs Income', fontsize=15)

plt.xlabel('Income (in $1000s)')

plt.ylabel('LotSize (in 1000 ft.sq.)')

plt.scatter(group1["Income"],group1["LotSize"], c="red", label='owner')

plt.scatter(group2["Income"],group2["LotSize"], c="green", label='non-owner')

plt.legend()

#Q3(a)

#Importing and reading excel file to dataframe

df\_AppShip=pd.read\_excel('/Users/ag30103/Desktop/GBUS738/ApplianceShipments.xlsx', delimiter=',', header=0, nrows=25, skiprows=0)

#Linear Time plot for series of quaterly shipments

fig2=plt.figure(figsize=(20, 5))

plt.xlabel('Quarter')

plt.ylabel('Shipments (in Million $)')

fig2.suptitle('Quarter vs Shipments', fontsize=15)

Quarter=df\_AppShip.iloc[:, 0]

Quarter=Quarter.str.split("-", n=1, expand=True)

Quarter=Quarter.iloc[:,1]+'-'+Quarter.iloc[:,0]

df\_AppShip.iloc[:,0]=Quarter

sorted\_df\_AppShip=df\_AppShip.sort\_values(by=['Quarter'])

plt.plot(sorted\_df\_AppShip["Quarter"],sorted\_df\_AppShip["Shipments"])

#Q3(b)

#Importing and reading excel file to dataframe

df\_AppShip=pd.read\_excel('/Users/ag30103/Desktop/GBUS738/ApplianceShipments.xlsx', delimiter=',', header=0, nrows=25, skiprows=0)

#Plotting time plot for Shipments based on four separate lines for four quarters

Quarter=df\_AppShip.iloc[:, 0]

Quarter=Quarter.str.split("-", n=1, expand=True)

df\_AppShip["Quarter"]=Quarter[0]

df\_AppShip["Year"]=Quarter[1]

df\_AppShip\_grouped=df\_AppShip.groupby("Quarter")

group1=df\_AppShip\_grouped.get\_group("Q1")

group2=df\_AppShip\_grouped.get\_group("Q2")

group3=df\_AppShip\_grouped.get\_group("Q3")

group4=df\_AppShip\_grouped.get\_group("Q4")

fig3=plt.figure(figsize=(15, 5))

plt.xlabel('Year')

plt.ylabel('Shipments (in Million $)')

fig3.suptitle('Year vs Shipments', fontsize=15)

plt.plot(group1["Year"],group1["Shipments"], c="red", label='Q1')

plt.plot(group2["Year"],group2["Shipments"], c="yellow", label='Q2')

plt.plot(group3["Year"],group3["Shipments"], c="green", label='Q3')

plt.plot(group4["Year"],group4["Shipments"], c="blue", label='Q4')

plt.legend()