CS 422: Data Mining

Department of Computer Science
Illinois Institute of Technology
Vijay K. Gurbani, Ph.D.

Fall 2021: Homework 5 (10 points)

Due date: Sat, October 16 2021, 11:59:59 PM Chicago Time

Please read all of the parts of the homework carefully before attempting any question. If you detect any ambiguities in the instructions, please let me know right away instead of waiting until after the homework has been graded.

Remember: This is an individual assignment. Sharing of code is strictly prohibited, and affected students will loose points if code is determined to have been shared.

1. Questions

1.1 (1 point) Tan, Chapter 3 Exercise 2, 3, 5.

1.2 (1 point) Tan, Chapter 4 Exercise 18. (Show your work, don't just provide the answer without showing how you derived it.)

2. Problems

2.1 (3 points) Exploratory data analysis on the Hotels Bookings dataset.

The hotels_bookings.csv dataset describes hotel demand data. One of the hotels is a resort hotel and the other is a city hotel. The dataset consists of 32 dimensions (the response variable is is_canceled) and 119,390 observations. Each observation represents a hotel booking, characterized by the dimensions described in the table below.

Perform exploratory data analysis and answer the questions below.

Attribute	Description
hotel	Resort Hotel (H1) or City Hotel (H2)
is_canceled	Value indicating if the booking was canceled (1) or not (0)
lead_time	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date
arrival_date_year	Year of arrival date
arrival_date_month	Month of arrival date
arrival_date_week_number	Week number of year for arrival date
arrival_date_day_of_month	Day of arrival date
stays_in_weekend_nights	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel
stays_in_week_nights	Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel
adults	Number of adults
children	Number of children

babies	Number of babies
meal	Type of meal booked. Categories are presented in standard hospitality meal packages: Undefined/SC – no meal package; BB – Bed & Breakfast; HB – Half board (breakfast and one other meal – usually dinner); FB – Full board (breakfast, lunch and dinner)
country	Country of origin. Categories are represented in the ISO 3155–3:2013 format
market_segment	Market segment designation. In categories, the term “TA” means “Travel Agents” and “TO” means “Tour Operators”
distribution_channel	Booking distribution channel. The term “TA” means “Travel Agents” and “TO” means “Tour Operators”
is_repeated_guest	Value indicating if the booking name was from a repeated guest (1) or not (0)
previous_cancellations	Number of previous bookings that were canceled by the customer prior to the current booking
previous_bookings_not_canceled	Number of previous bookings not canceled by the customer prior to the current booking
reserved_room_type	Code of room type reserved. Code is presented instead of designation for anonymity reasons.
assigned_room_type	Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons.
booking_changes	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation
deposit_type	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories: No Deposit – no deposit was made; Non Refund – a deposit was made in the value of the total stay cost; Refundable – a deposit was made with a value under the total cost of stay.
agent	ID of the travel agency that made the booking.
company	ID of he company/entity that made the booking or responsible for paying the booking. ID is presented instead of designation for anonymity reasons
days_in_waiting_list	Number of days the booking was in the waiting list before it was confirmed to the customer
customer_type	Type of booking, assuming one of four categories: Contract - when the booking has an allotment or other type of contract associated to it; Group – when the booking is associated to a group; Transient – when the booking is not part of a group or contract, and is not associated to other transient booking; Transient-party – when the booking is transient, but is associated to at least other transient booking
adr	Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights
required_car_parking_spaces	Number of car parking spaces required by the customer
total_of_special_requests	Number of special requests made by the customer (e.g. twin bed or high floor)
reservation_status	Reservation last status, assuming one of three categories: Canceled – booking was canceled by the customer; Check-Out – customer has checked in but already departed; No-Show – customer did not check-in and did inform the hotel of the reason why
reservation_status_date	Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the customer

(a) How many observations exist in the dataset for hotel type H1 and hotel type H2? Use R commands to print the frequency of the hotel types. (Hint: Do not write a loop, one R command will do this.)

(b) What is the distribution of the class label in the dataset? Your output should look like the following:

```
Number of guests who canceled reservation: XXXX
Number of guests who did not cancel the reservation: XXXX
```

(c) Which customer has the most reservations? Your output should look like the following:

```
Customer type with the most reservations is XXXX, with YYYY reservations
```

(Hint: Use the command from (a), and play around with which.max(), and use cat() and paste() to create the above output string).

(d) What was the **most** number of parking spaces required by customers? And how many customers requested that many parking spaces? Your output should look like the following, where XXXX is the number of customers and YYYY is the number of parking spaces:

```
XXXX customers required the most number of parking spaces (YYYY).
```

(Hint: Use the same hint from (c)).

(e) What was the **least** number of parking spaces required by customers? And how many customers requested that many parking spaces? Your output should look like the following, where XXXX is the number of customers and YYYY is the number of parking spaces:

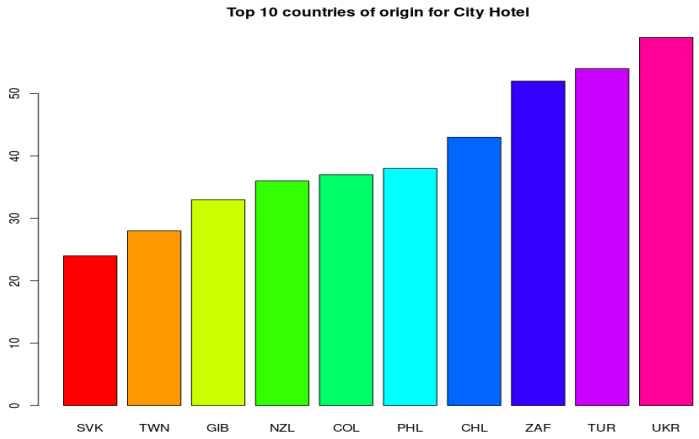
```
XXXX customers required the least number of parking spaces (YYYY).
```

(Hint: Use the same hint from (c)).

(f) How many people who expressed a preference for a particular room type during reservation were actually assigned that specific room type? Your output should look like the following:

```
XX.XX% of the people who expressed a room preference during reservation got the
room during check-in.
```

(g) Order the dataset between city hotels and resort hotels. For each type of hotel, find the top 10 countries of origin that attract the most bookings. Plot a bar plot for each hotel type that contains the country of origin. Make sure you use colors for each bin and title each plot accordingly. The x-axis of the barplot should contain the 3- (or 2-) letter code for the country. At the very least, your graphical output should look like the following, but try to make it as attractive as possible.



Hints: (1) To sort, you should not write a loop. The sorting and the plotting can be accomplished in 3 lines of code at the maximum. (2) To plot, see barplot() API. (3) When you get the top 10 countries of origin for Resort Hotels, you will notice an anomaly. Do not plot the barplot containing that anomaly, and instead, remove the anomaly and plot the country previous to where the anomaly occurs.

(h) You will note that the most visitors to either type of the hotels arrive from a specific country.
(i) Print the name of this country.
(ii) What can you say about the origin of the dataset based on (i)?

2.2 Decision Tree (3 points)

You will use the same dataset as in Problem 2.1.

Using the rpart package in R, train a decision tree to predict whether a booking will be canceled or not.

Use a seed of 1122 to randomly assign 90% of the dataset to training, and 10% to testing.

(a) **(2 points)** Create the **best** decision tree model that you can to predict whether a booking will be canceled or not. You may use as many (or as least) number of predictor variables, it is up to you. Hint: Think!
(i) Plot the decision tree.
(ii) List which variables are important.
(iii) Fit the model on the held out test dataset, and from the resulting confusion matrix, print the following attributes: Accuracy, Error, Balanced Accuracy, Specificity, Sensitivity, and Precision.
(iv) Plot a ROC curve on your held out test dataset. (Make sure you use library(ROCR) to use the ROC-specific APIs.)
(v) What is the AUC of the ROC curve.

I will run (a) as a contest among all students. Email me and the TA your **best** attempted scores in (iii) using the following format:

A#,0.78,0.22,0.70,0.87,0.56

where the first field is your A#, the second is Accuracy, the third is Error, etc., for all the fields in (iii). From now until the end of the submission period for this homework, we will post your best attempt publicly in the a leader scoreboard. We will publish the URL to this scoreboard so each student can see how his or her peers are doing. If you build a model that betters your score, send it to us and we will update your entry on the leader scoreboard.

The leaderboard URL is http://www.cs.iit.edu/~vgurbani/leaderboard.txt

2.3 (2 points) Having fun with pruning!

In this problem, you will use the same dataset you used in Problem 2.1.

Set seed to 1122 randomly assign 90% of the dataset to training, and 10% to testing.

(a) In Problem 2.2 you created the best decision tree using a certain number of predictors. Using the same predictors, re-train the model on the training dataset except in this problem, you will set the complexity parameter to 0.0. (See the manual page for rpart() and figure out how to set the complexity parameter. Hint: look for the control parameter.) Do not plot the model, as the tree is too deep to be plotted.

Using your model, predict the held-out dataset and print the following information in the format shown below (**round all numbers to 3 decimal places**):

```
Before pruning:
  Accuracy: X.XXX
  Error: X.XXX
  Balanced Acc.: X.XXX
  Specificity: X.XXX
```

```
  Sensitivity: X.XXX
  Precision: X.XXX
```

(b) Now, prune the tree using the prune() method. Find the point in the tree that you want to prune, and print out the following (**round all numbers to 5 decimal places**):

```
Prune point occurs at a complexity of X.XXXXX
At this complexity, xerror is X.XXXXX
```

(c) Using this pruned tree, predict the held-out dataset and print the following information in the format shown below (**round all numbers to 3 decimal places**):

```
After pruning:
  Accuracy: X.XXX
  Error: X.XXX
  Balanced Acc.: X.XXX
  Specificity: X.XXX
  Sensitivity: X.XXX
  Precision: X.XXX
```

(d) Which model --- the full tree, or the pruned tree --- generalizes better. Your output should be of the form:

The _____ tree generalizes better.

(e) In Problem 2.2, you created the best decision tree model. Which model among the three models --- (1) model in Problem 2.2, (2) model in Problem 2.3(a), (3) model in Problem 2.3(b) --- generalizes the best?