Research Skills: Programming with R

Assignment 2

This graded set of homework assignments must be handed in on Canvas before Friday, December 13th, 21:00. It tests your mastery of Worksheets 4 to 6. You will be asked to write functions and apply them repeatedly, to clean and tidy data sets, and to fit and evaluate classification models. NOTE: consult the ppt for Class 7 for additional tips.

It will be graded as follows:

- 0.5 point each for Questions 1 through 7
- 1.5 point each for Questions 8 through 10
- 1.0 point in total for overall code organisation & style
- 1.0 point in total for complying with the instructions below

The guidelines for overall code organisation & style can be found in the slides for Class 4. Note that to receive full marks for this aspect you will have to make use of the `%>%` operator where applicable.

Questions 1 through 7 can be graded semi-automatically. All correct solutions will receive full marks, and any deviations from the requested answers, down to misspellings, will receive 0 points. For Questions 8 through 10, partial solutions will receive partial points, and the efficiency and succinctness of your answers will matter.

All questions are independent except for Question 10; copy the data set before modifying it, and start afresh with the original each time. Other instructions:

- solve all the questions in a single R script
- use `Programming_with_R_2019_BLOCK2_A2-script_template.R`, from Canvas, as the basis of this script
- load the data exactly as shown in this demo; do not adapt the relative paths
- use any function from 'base R', `dplyr`, `tidyr`, `ggplot2`, `caret`, and no other packages
- name your script `Lastname_U-number_Assignment2.R`
- include your name and u-number at the top of your script
- store your solutions to Questions 1 - 7 in the objects described

This is an individual assignment: You may discuss it with your fellow students in general terms but do not share code. Evidence of plagiarism will be referred to the Exam Committee. Good luck!

Data Set Information

This assignment uses three artificial data sets. The first, `mushrooms`, is a heavily edited version of the `mushrooms` data set available from the UCI Machine Learning Repository. It contains ~8000 observations of 23 made-up mushroom species. The second, `edibility`, classifies each made-up species as edible or poisonous. The final data set, `survey`, is a made-up record of mushroom counts taken in a specific survey area throughout the year.

Question 1.

Create a function which accepts as its arguments a dataframe and a string. You may assume that the dataframe is the `mushrooms` data set or a subset of it, and that the string is a `habitat`. The function should return the number of times the specified habitat occurs in the dataframe. Create this function with a meaningful name initially, then store it in an object called `answer1`. It should then be possible to call it like this:

```
answer1(mushrooms, "leaves")
```

```
## [1] 832
```

Question 2.

Create a copy of the `mushrooms` data set which includes an additional column, `white_parts`. This column should report the total number of white parts for each observation in the data set, i.e., how many of `cap_color`, `gill_color`, `stalk_color`, `veil_color`, and `spore_print_color` are equal to `"white"`. Create this dataframe with a meaningful name initially, then store it in an object called `answer2`.

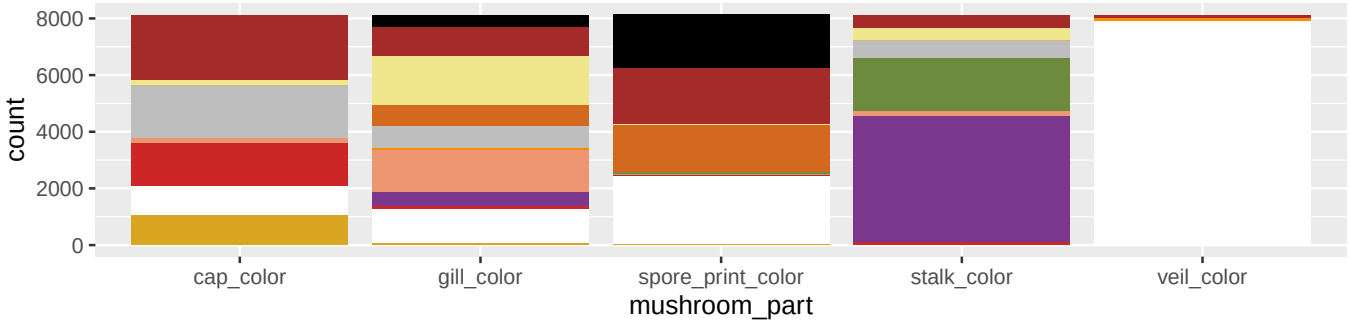
Question 3.

Create two new copies of the `mushrooms` data set: The first should use the `edibility` data set to add a new column, `edibility`, which specifies for each observation whether the mushroom described is edible. The second should use the `survey` data set to exclude all observations of species that do not occur at all in the area surveyed. Create these objects with meaningful names initially, then store them in objects called `answer3_a` and `answer3_b`, respectively.

Question 4.

For the code below to create the plot shown, the `mushrooms` dataframe must be re-shaped first. Create the appropriately re-shaped dataframe with a meaningful name initially, then store it in an object called `answer4`. This object should deliver the plot as shown, using the exact code shown below.

```
ggplot(answer4, aes(x = mushroom_part, fill = color_of_part)) +
  geom_bar() +
  scale_fill_manual(values = c("white" = "white", "yellow" = "goldenrod",
    "pink" = "lightsalmon2", "buff" = "khaki", "brown" = "brown",
    "gray" = "gray", "black" = "black", "green" = "darkolivegreen4",
    "purple" = "mediumorchid4", "red" = "firebrick3", "cinnamon" = "tan",
    "orange" = "darkorange", "chocolate" = "chocolate"), guide = FALSE)
```



(Note: For this question, you thus only need to store the re-shaped data set in your `answer4` object; using this `answer4`, it should be possible to produce the plot shown without any alterations to the provided plot code at all.)

Question 5.

In addition to `clean_survey.txt`, there's also a `raw_survey.txt` attached to this assignment. Imagine that it's another year's raw `survey` results. Read it into R, ensure that the column names are correct, and fix any obviously wrong inputs. Once you are finished, it should resemble the `survey` data set in every way except for the exact monthly counts. Create this object with a meaningful name initially, then store it in an object called `answer5`.

Question 6.

Using the `mushrooms` data set and `train()`, fit a `"knn"` model using 3-fold cross validation, optimising accuracy. It should predict `species` based on all other variables; try values for `k` of 3, 5 and 7. Use `set.seed(1)` before fitting this model. Create it with a meaningful name initially, then store it in an object called `answer6`.

(Note: For this question, you thus do not need to split `mushrooms` into a train -and test set.)

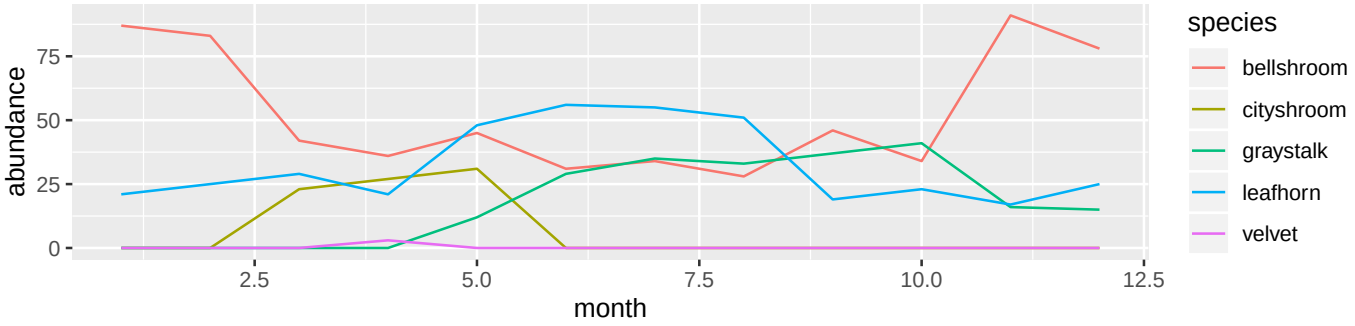
Question 7.

Using the `mushrooms` data set and `train()`, fit a logistic regression model using 3-fold cross validation, optimising accuracy. It should predict `bruises` based on `cap_color`, `odor`, and their interaction. Make sure that the reference level for `odor` is `"none"`; otherwise stick to the defaults. Use `set.seed(1)` before fitting this model. Create it with a meaningful name initially, then store it in an object called `answer7`.

(Note: For this question, you thus do not need to split `mushrooms` into a train -and test set. Ignore any warnings.)

Question 8.

Using the `survey` and `edibility` data sets, create a line graph showing the number of edible mushrooms found in the survey area each month. Each edible species should be represented by its own line, and the months should be arranged chronologically on the x-axis, from January to December. (The actual labels on the x-axis do not matter; see one possible plot below for inspiration, but note that you do not need to replicate this plot's aesthetics exactly).



Question 9.

Create a copy of the `mushrooms` data set that is re-formatted in the following ways:

- all the entries in all the columns concerning the cap, gills, and stalk should have `cap`, `gills` or `stalk` appended to them, respectively; separate the original entry from the specification using a `.`; note that the `gills` columns use `gill` in their names, so you'll have to add an extra *s*
- all entries and all column names should be entirely capitalised
- all multiple-word column names should be split by dots, not underscores

The first few rows of a correct solution should look like this:

```
##      SPECIES  CAP.SHAPE CAP.SURFACE  CAP.COLOR BRUISES    ODOR GILL.ATTACHMENT
## 1  PUNGENTIA CONVEX.CAP   SCALY.CAP  BROWN.CAP     YES PUNGENT      FREE.GILLS
## 2  YELLOWCAP CONVEX.CAP   SCALY.CAP YELLOW.CAP     YES  ALMOND      FREE.GILLS
## 3 BELLSHROOM   BELL.CAP   SCALY.CAP  WHITE.CAP     YES   ANISE      FREE.GILLS
##   GILL.SPACING    GILL.SIZE  GILL.COLOR     STALK.SHAPE  STALK.ROOT
## 1  CLOSE.GILLS NARROW.GILLS BLACK.GILLS ENLARGING.STALK EQUAL.STALK
## 2  CLOSE.GILLS  BROAD.GILLS BLACK.GILLS ENLARGING.STALK  CLUB.STALK
## 3  CLOSE.GILLS  BROAD.GILLS BROWN.GILLS ENLARGING.STALK  CLUB.STALK
##   STALK.SURFACE  STALK.COLOR VEIL.COLOR RING.NUMBER RING.TYPE SPORE.PRINT.COLOR
## 1  SMOOTH.STALK PURPLE.STALK      WHITE         ONE   PENDANT             BLACK
## 2  SMOOTH.STALK PURPLE.STALK      WHITE         ONE   PENDANT             BROWN
## 3  SMOOTH.STALK PURPLE.STALK      WHITE         ONE   PENDANT             BROWN
##   POPULATION HABITAT
## 1  SCATTERED   URBAN
## 2   NUMEROUS GRASSES
## 3   NUMEROUS MEADOWS
```

Question 10.

Split `answer3_a`, which adds edibility information to the `mushrooms` data set, into a train and test set. 80% of the observations should be in the train set, 20% in the test set; the `edibility` variable should be balanced between the splits. Fit a `"knn"` model predicting `edibility` on the basis of all other variables, optimizing *recall*. Use 5-fold cross validation, and test the default values of k. Then create a `confusionMatrix()` for the test set, and for your final answer, extract only "Precision" and "Recall" from it. Use `set.seed(1)` at the very start of your solution.

If you were unable to solve `answer3a`, use the code below to create a `backup_shrooms` object to work with instead.

```
edible_shrooms <- sample(edibility$species, 4)
backup_shrooms <- mushrooms
backup_shrooms$edibility = "poisonous"
backup_shrooms$edibility[backup_shrooms$species %in% edible_shrooms] <- "edible"
```