Week 2 Classwork/Homework Assignment Answers

John Vinson

August 21, 2017

For the assignment, you will submit a R Markdown (.rmd) file and the compiled document as both a pdf and html. Make sure that the compiled documents will display all the required code to get your results.

The homework assignment is due August 25, 2017.

1. Write and execute a chunk of code that performs each of these operations. For each line of code, annotate it to describe what it will do when run.

- Print the line "Here are my answers for Question 1."
- Compute the product of six and five.
- Compute eight to the fifth power.
- Compute the product of two variables, A and B, where A is nine and B is eighty.

```
print("Here are my answers for Question 1.") #print the line "Here are my answers for Question 1."
```

```
## [1] "Here are my answers for Question 1."
```

```
6*5 #calculates the product of 8 and 5
```

```
## [1] 30
```

```
8^5 #calculates 8 raised to the 5th power
```

```
## [1] 32768
```

```
A=9 #initialize the variable A
B=80 #initialize the variable B

A*B #calculates the product of variables A and B
```

```
## [1] 720
```

2. Use the following set to complete the following operations.

$$\{13.0, 1.0, 5.0, 8.0, 1.0, 0.0, 34.0, 2.0, 21.0, 0.0, 3.0\} \quad (1)$$

- Create a vector named *Ornacia* using the set.
- Determine the length of the vector.
- Find the position of value 21.
- Find the highest values and the positions of the value(s).
- Find the lowest value(s). Display the lowest values(s).
- Create a new vector, *Ornacia.Order*, which is the vector ordered from the lowest to highest numbers.
- Create a new vector, *Ornacia.Even*, which are only the values of the even positions in the vector.
- Append the following numbers to the end of the vector *Ornacia* $\{55.0, 89.0\}$
- Create a new vector, *Ornacia.Half*, which contains all half of all the values of *Ornacia*.

```
Ornacia = c(13.0,1.0,5.0,8.0,1.0,0.0,34.0,2.0,21.0,0.0,3.0)

length(Ornacia)
```

```
## [1] 11
```

```
max(Ornacia)
```

```
## [1] 34
```

```
which(Ornacia == max(Ornacia))
```

```
## [1] 7
```

```
min(Ornacia)
```

```
## [1] 0
```

```
which(Ornacia == min(Ornacia))
```

```
## [1]  6 10
```

```
which(Ornacia == 21)
```

```
## [1] 9
```

```
Ornacia.Order = Ornacia[order(Ornacia)]
Ornacia.Order
```

```
##  [1]  0  0  1  1  2  3  5  8 13 21 34
```

```
Ornacia.Even = Ornacia[seq(2, length(Ornacia), by=2)]
Ornacia.Even
```

```
## [1] 1 8 0 2 0
```

```
Ornacia = c(Ornacia, c(55.0, 89.0))
#Or you could do
#Ornacia = append(Ornacia, c(55.0, 89.0))
Ornacia
```

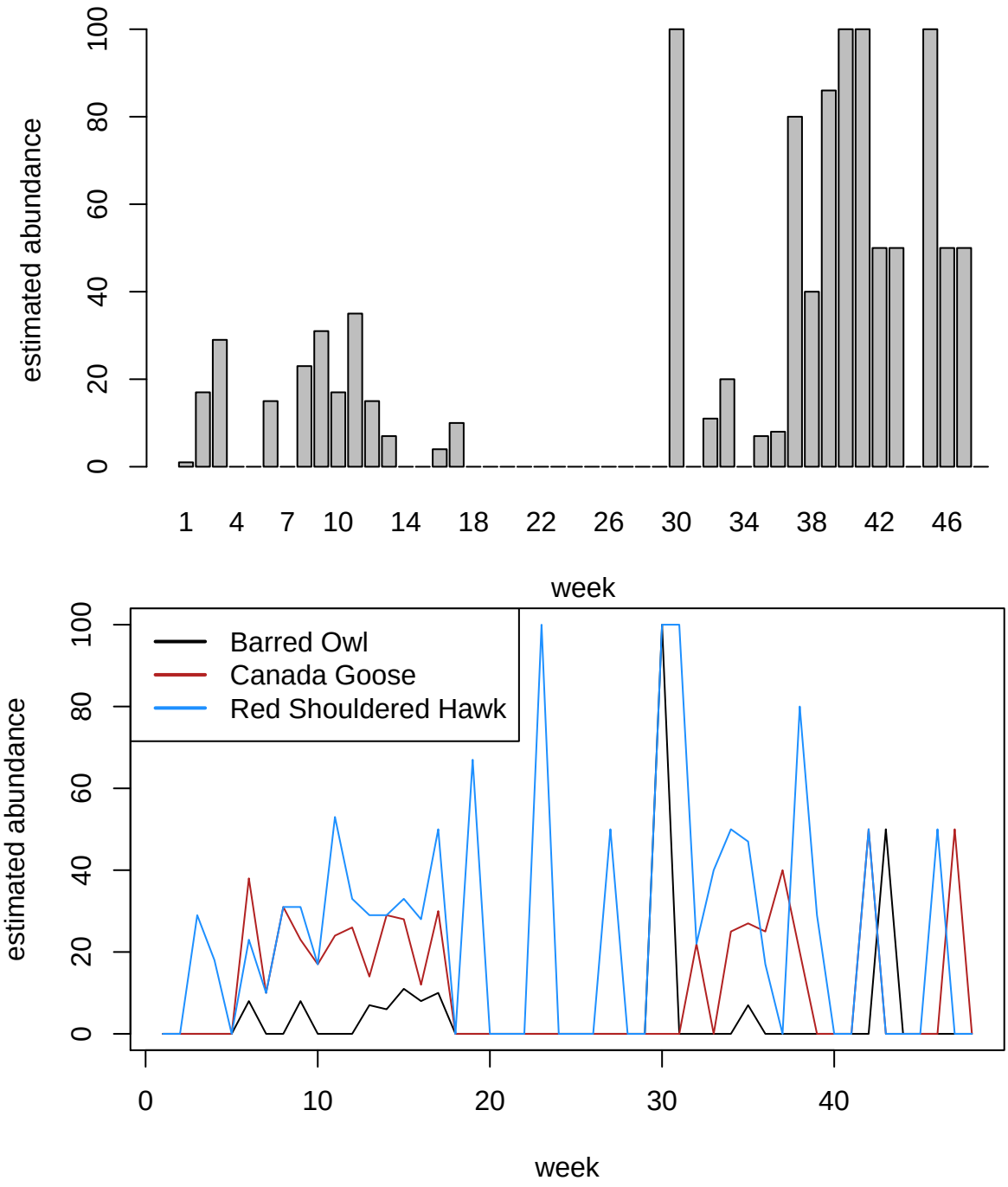
```
##  [1] 13  1  5  8  1  0 34  2 21  0  3 55 89
```

```
Ornacia.Half = Ornacia/2
Ornacia.Half
```

```
##  [1]  6.5  0.5  2.5  4.0  0.5  0.0 17.0  1.0 10.5  0.0  1.5 27.5 44.5
```

3. Import the dataset "eBird_BotGarden_2016.csv". These are estimated abundances of birds seen at the State Botanical Gardens of Georgia (Athens, GA) each week of 2016. Perform the following operations:

- Plot the weekly abundance of the Northern Flicker as a bar graph.
- Plot the weekly abundance of the Barred Owl, Canada Goose and Red Shouldered Hawk on a single line graph each with a different color. Create a legend for each of the species.



4. Using the eBird data (from question 3), write a function that will calculate the average abundance of a specified species for the year. Your function should have two arguments/inputs: the dataset and the name of the species. The output should be a single number (the average). Find the average abundance for the Canada Goose. (Hint: The *names* function will return the column names of a data frame.)

```
averaging.fun = function(data, species){
  species.names = names(data)
  col.need = which(species.names==species)
  return(mean(data[,col.need]))
}
```

```
averaging.fun(ebird.data, "Canada.Goose")
```

```
## [1] 11.27083
```

5. Using your averaging function, write script to create a new dataframe containing the names and average abundance of every species that has an average abundance greater than 5 individuals. Your resulting data frame should have two columns: the name of the species and their average abundance.

```
avg.abun = NULL
species.keep = NULL

for(i in names(ebird.data)){
  if(i != "X" && i != "Week") {
    avg.temp = averaging.fun(ebird.data, i)
    if(avg.temp>5){avg.abun = c(avg.abun, avg.temp)
      species.keep = c(species.keep, i)
      }
  }
}

ebird.avg.data = data.frame(species.keep, avg.abun)

ebird.avg.data
```

```
##                 species.keep  avg.abun
## 1               Canada.Goose 11.270833
## 2                  Wood.Duck  7.145833
## 3           Great.Blue.Heron  6.520833
## 4              Black.Vulture 20.895833
## 5            Turkey.Vulture 33.479167
## 6           Mississippi.Kite 11.270833
## 7        Red.shouldered.Hawk 25.333333
## 8          Broad.winged.Hawk  5.270833
## 9            Red.tailed.Hawk 16.979167
## 10               Rock.Pigeon 12.812500
## 11             Mourning.Dove 49.958333
## 12      Yellow.billed.Cuckoo 20.708333
## 13             Chimney.Swift  7.541667
## 14 Ruby.throated.Hummingbird 18.895833
## 15    Red.bellied.Woodpecker 65.750000
## 16  Yellow.bellied.Sapsucker 18.270833
## 17          Downy.Woodpecker 58.833333
## 18          Hairy.Woodpecker 14.583333
## 19          Northern.Flicker 22.000000
## 20       Pileated.Woodpecker 19.916667
## 21        Eastern.Wood.Pewee 20.395833
## 22        Acadian.Flycatcher 25.500000
## 23            Eastern.Phoebe 48.437500
## 24  Great.Crested.Flycatcher 13.333333
```