UWDS2-Wk1-Albacore-jms206

Jim Stearns, NetID=jms206

13 January 2015

Albalone Data

Assignment Overview

"We used summary statistics, histograms, and box plots to explore the Abalone data in class. Now we'll use aggregation to continue to explore that dataset. We know the height of each abalone, but we can more clearly identify some trends if we map the large number of unique heights to just a few height groups. We'll also take a look at another kind of aggregation (averaging) to reveal a relationship between weight and age. Some of the solutions to this problem are contained in the slides we didn't have time to cover in the first lecture. Please try them on your own first."

Dataset Acquisition and Preparation

The Albalone dataset of interest can be found at the University of California Irvine dataset repository. Its header page states that there are 4177 instances/observations/rows and 8 attributes.

The actual dataset contains 9 columns. The description of attributes on the header page contains 9 attributes. None of the fields is a unique identifier that clearly should be excluded from analysis. The last column, Rings, is "the value to predict".

```
Given is the attribute name, attribute type, the measurement unit and a
brief description.  The number of rings is the value to predict: either
as a continuous value or as a classification problem.

 Name        Data Type   Meas.   Description
 ----        ---------   -----   -----------
 Sex     nominal         M, F, and I (infant)
 Length      continuous  mm  Longest shell measurement
 Diameter    continuous  mm  perpendicular to length
 Height      continuous  mm  with meat in shell
 Whole weight    continuous  grams   whole abalone
 Shucked weight  continuous  grams   weight of meat
 Viscera weight  continuous  grams   gut weight (after bleeding)
 Shell weight    continuous  grams   after being dried
 Rings       integer         +1.5 gives the age in years

Statistics for numeric domains:

     Length  Diam    Height  Whole   Shucked Viscera Shell   Rings
 Min 0.075   0.055   0.000   0.002   0.001   0.001   0.002       1
 Max 0.815   0.650   1.130   2.826   1.488   0.760   1.005      29
```

```
    Mean    0.524   0.408   0.140   0.829   0.359   0.181   0.239   9.934
    SD  0.120   0.099   0.042   0.490   0.222   0.110   0.139   3.224
    Correl  0.557   0.575   0.557   0.540   0.421   0.504   0.628     1.0
```

The dataset does not contain a header row. Add appropriate column headers.

```
urlUciAboloneData <- "http://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data"
abo <- read.csv(urlUciAboloneData, header=FALSE)
abo_names=c("Sex","Length", "Diameter","Height", "Whole_weight", "Shucked_weight",
    "Viscera_weight","Shell_weight","Rings")
names(abo) <- abo_names
```

Put the data.frame into a data.table in order to simplify column references below.

```
library(data.table)
abodt <- data.table(abo)
summary(abodt)
```

```
##  Sex          Length         Diameter          Height
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300
##   Whole_weight    Shucked_weight   Viscera_weight    Shell_weight
##  Min.   :0.0020   Min.   :0.0010   Min.   :0.0005   Min.   :0.0015
##  1st Qu.:0.4415   1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300
##  Median :0.7995   Median :0.3360   Median :0.1710   Median :0.2340
##  Mean   :0.8287   Mean   :0.3594   Mean   :0.1806   Mean   :0.2388
##  3rd Qu.:1.1530   3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290
##  Max.   :2.8255   Max.   :1.4880   Max.   :0.7600   Max.   :1.0050
##      Rings
##  Min.   : 1.000
##  1st Qu.: 8.000
##  Median : 9.000
##  Mean   : 9.934
##  3rd Qu.:11.000
##  Max.   :29.000
```

Deciles

Assignment

a. "Use the command 'quantile' to find the deciles (10 groups) for height from the complete data set. Hint: you may find the command "seq" helpful."

Implementation

```
heightDecile <- quantile(abodt$Height, probs=seq(0, 1, 0.1))
heightDecile
```

```
##    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
## 0.000 0.090 0.105 0.120 0.130 0.140 0.150 0.160 0.175 0.185 1.130
```

Note: values in vector (Height) need not be sorted. Quantile will sort.

Sanity check: First quartile value is between 20% and %30; third quartile is between 70% and 80%.

Age vs Height Deciles

Assignment

b. "Use the command "cut" to assign each height value to the corresponding decile (e.g., the smallest values are assigned to the first decile and get mapped to the value, 1). Hint: use "as.numeric" to get integer values instead of ranges."
c. "Now create a table of age vs. height decile. Examine the table and describe what you observe."

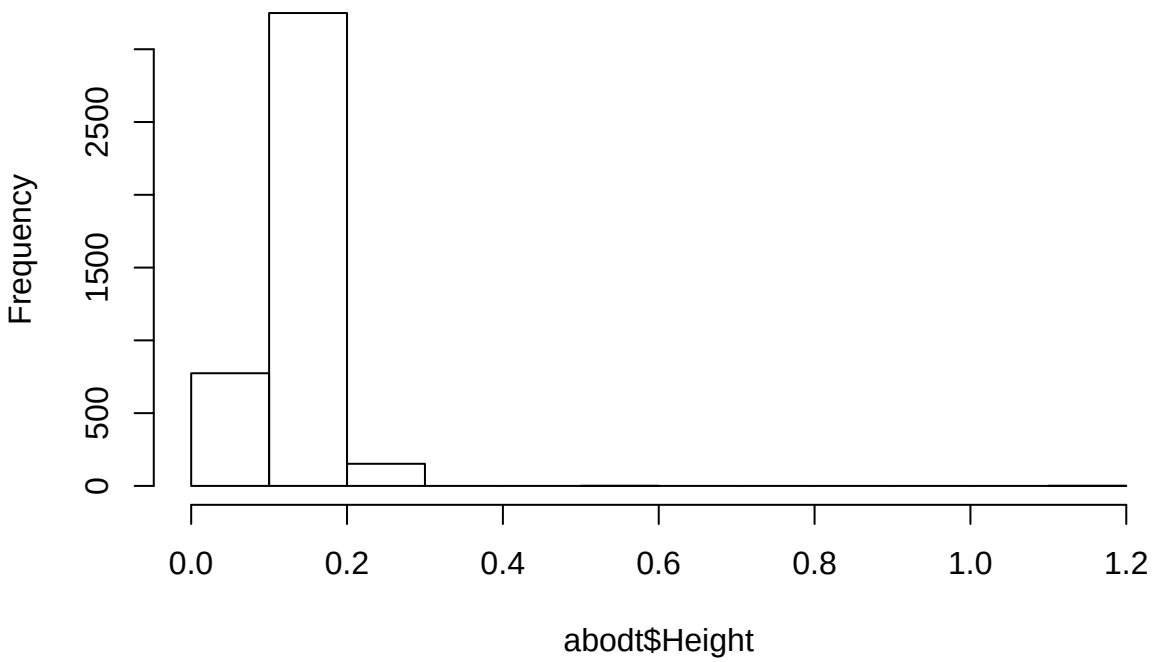
Implementation

The dataset does not contain an age column. However, Rings provides the data for calculating age: the description for Rings says that "+1.5 gives the age in years".

```
abodt$Age <- abodt$Rings + 1.5
```

Heights are mostly in the first two deciles the value range, with a few "highliers":

```
hist(abodt$Height)
```




```
cuts <- as.numeric(cut(sort(abodt$Height), 10))
table(cuts)
```

```
## cuts
##    1    2    3    5   10
## 1023 3129   23    1    1
```

Age decile vs height decile:

```
ageDecile <- quantile(abodt$Age, probs=seq(0, 1, 0.1))
table(ageDecile, heightDecile)
```

```
##           heightDecile
## ageDecile 0 0.09 0.105 0.12 0.13 0.14 0.15 0.16 0.175 0.185 1.13
##      2.5  1    0     0    0    0    0    0    0     0     0    0
##      7.5  0    1     0    0    0    0    0    0     0     0    0
##      8.5  0    0     1    0    0    0    0    0     0     0    0
##      9.5  0    0     0    1    0    0    0    0     0     0    0
##      10.5 0    0     0    0    1    1    0    0     0     0    0
##      11.5 0    0     0    0    0    0    1    0     0     0    0
##      12.5 0    0     0    0    0    0    0    1     0     0    0
##      13.5 0    0     0    0    0    0    0    0     1     0    0
##      15.5 0    0     0    0    0    0    0    0     0     1    0
##      30.5 0    0     0    0    0    0    0    0     0     0    1
```

Roughly linear. Positively correlated (0.94)

Average Weight as Function of Age

Assignment

d. "Another way to aggregate the data is averaging. Let's compute the average whole weight of abalone as a function of age and plot the relationship."

e. "Use the commands "unique" and "sort" to find the unique values of Age and store the values in ascending order to a variable named "ua"'."

ii. "Use the command "sapply" to apply a function to each value in "ua". The function should return the mean whole weight of all abalone of a given age. Hint: type "help('function')" to find out more about user defined functions. The quotes inside the parentheses are important."

iii. "Finally, use the "plot" command to plot mean weight vs. age. Describe the relationship revealed by the plot. Include an explanation for the behavior seen in the abalone of the 25-30 year age group."

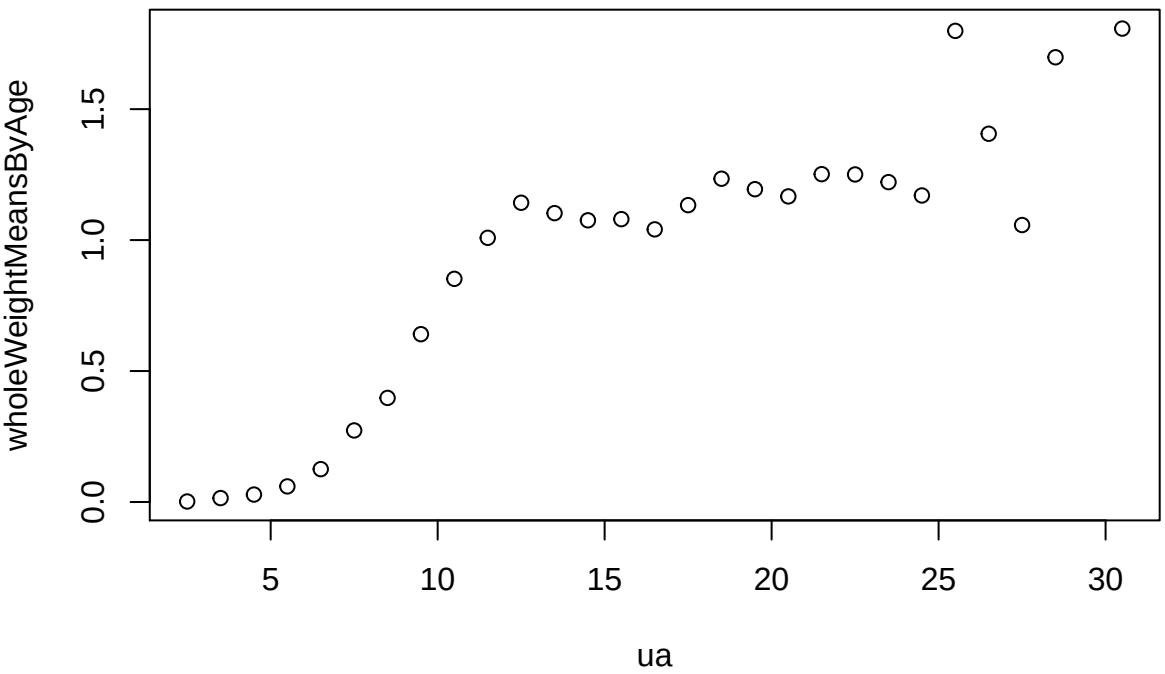
Implementation

```
ua <- sort(unique(abodt$Age))

wholeWeightMean <- function(age, abo_dt) {
    mean(abo_dt[Age == age, Whole_weight])
}

wholeWeightMeansByAge <- sapply(ua, wholeWeightMean, abodt)

plot(ua, wholeWeightMeansByAge)
```



Weight grows with age until plateauing around 13 years. Weight means above 25 years scatter because the population at that age is very small (7 out of a population of 4177).