Assignment #4 - Sample Solution

Problem 1: Naive Bayes Classifier

Problem 1(a): Category probabilities

For each of the three categories for Goal -- Grades, Popular, Sports -- compute the probability of that category, i.e., the fraction of the Schoolkids data items in that category. Your answer should consist of one number for each category, with the three numbers summing to 1.0. Don’t forget to state what tool you used to compute your answer, and demonstrate how the tool was used.

Answer:

Grades 0.52
Popular 0.29
Sports 0.19

Used Google Sheets pivot table with formula in column C:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grades</td>
<td>247</td>
<td>0.5167364017</td>
</tr>
<tr>
<td>Popular</td>
<td>141</td>
<td>0.2949790795</td>
</tr>
<tr>
<td>Sports</td>
<td>90</td>
<td>0.1882843188</td>
</tr>
<tr>
<td>Grand Total</td>
<td>478</td>
<td></td>
</tr>
</tbody>
</table>

Problem 1(b): Conditional probabilities within categories

Now separately consider each of the three categories for Goal. For each one, compute the probabilities within that category for each of the possible values for each of the three features -- Gender, Grade, and Type. For each category, you should have 8 numbers (so your answer will consist of 24 numbers total): the fraction of girls versus boys within that category (summing to 1.0); the fraction of grades 4, 5, and 6 within that category (summing to 1.0); the fraction of Rural, Suburban, and Urban within that category (summing to 1.0). Again, don’t forget to state what tool you used to compute your answer, and demonstrate how the tool was used.

Answer:

Grades
- boy 0.47
- girl 0.53
- grade 4 0.25
- grade 5 0.36
- grade 6 0.39
- Rural 0.23
- Suburban 0.35
- Urban 0.42
Popular
- boy 0.35
- girl 0.65
- grade 4 0.22
- grade 5 0.39
- grade 6 0.39
- Rural 0.35
- Suburban 0.3
- Urban 0.35

Sports
- boy 0.67
- girl 0.33
- grade 4 0.28
- grade 5 0.37
- grade 6 0.35
- Rural 0.47
- Suburban 0.24
- Urban 0.29

Used Google Sheets pivot tables with formula in columns E,F,G,H:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>boy</td>
<td>girl</td>
<td>Grand Total</td>
<td>boyfraction</td>
<td>girlfraction</td>
<td></td>
</tr>
<tr>
<td>Grades</td>
<td>117</td>
<td>130</td>
<td>247</td>
<td>0.4736842105</td>
<td>0.5263157895</td>
<td></td>
</tr>
<tr>
<td>Popular</td>
<td>50</td>
<td>91</td>
<td>141</td>
<td>0.3546099291</td>
<td>0.645390709</td>
<td></td>
</tr>
<tr>
<td>Sports</td>
<td>60</td>
<td>30</td>
<td>90</td>
<td>0.6666666667</td>
<td>0.333333333</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>Grand Total</td>
<td>4 fraction</td>
<td>5 fraction</td>
<td>6 fraction</td>
</tr>
<tr>
<td>Grades</td>
<td>63</td>
<td>88</td>
<td>96</td>
<td>247</td>
<td>0.2550607287</td>
<td>0.3562753036</td>
<td>0.3886639676</td>
</tr>
<tr>
<td>Popular</td>
<td>31</td>
<td>55</td>
<td>55</td>
<td>141</td>
<td>0.219658156</td>
<td>0.390070922</td>
<td>0.390070922</td>
</tr>
<tr>
<td>Sports</td>
<td>25</td>
<td>33</td>
<td>32</td>
<td>90</td>
<td>0.2777777778</td>
<td>0.3666666667</td>
<td>0.3555555556</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rural</td>
<td>Suburban</td>
<td>Urban</td>
<td>Grand Total</td>
<td>Rural fraction</td>
<td>Suburban fraction</td>
<td>Urban fraction</td>
</tr>
<tr>
<td>Grades</td>
<td>57</td>
<td>87</td>
<td>103</td>
<td>247</td>
<td>0.2307692308</td>
<td>0.3522267206</td>
<td>0.4170049486</td>
</tr>
<tr>
<td>Popular</td>
<td>50</td>
<td>42</td>
<td>49</td>
<td>141</td>
<td>0.3546099291</td>
<td>0.2978723404</td>
<td>0.3475177305</td>
</tr>
<tr>
<td>Sports</td>
<td>42</td>
<td>22</td>
<td>26</td>
<td>90</td>
<td>0.4666666667</td>
<td>0.2444444444</td>
<td>0.2888888889</td>
</tr>
</tbody>
</table>
Problem 1(c): Category assignment

Now you’re ready to predict a category for new items. Suppose new Item #1 is a 5th grade girl from a Suburban school, and Item #2 is a 4th grade boy from a Rural school. Using the probabilities from Steps 1 and 2, compute the most likely categories for Items #1 and #2. Show your calculations!

**Answer:**

**Item #1:**

Grades: $0.52 \times 0.53 \times 0.36 \times 0.35 = 0.0347$ - most likely  
Popular: $0.29 \times 0.65 \times 0.39 \times 0.3 = 0.022$  
Sports: $0.19 \times 0.33 \times 0.37 \times 0.24 = 0.00557$

**Item #2:**

Grades: $0.52 \times 0.47 \times 0.25 \times 0.23 = 0.014$  
Popular: $0.29 \times 0.35 \times 0.22 \times 0.35 = 0.0078$  
**Sports: $0.19 \times 0.67 \times 0.28 \times 0.47 = 0.01675$ - most likely**

Incidentally, the independent probability of Item #1’s features (grade=5 x gender=girl x Type=Suburban) over the entire dataset is 0.061, and the independent probability of item #2’s features (grade=4 x gender=boy x Type=Rural) over the entire dataset is 0.037. If we do the final, but unnecessary, Naive Bayes calculation of dividing by feature probabilities, we get actual probabilities for the three categories, adding up to 1.0 (well, close to 1.0, with compounding rounding errors):

**Item #1:**

Grades: $0.52 \times 0.53 \times 0.36 \times 0.35 = 0.0347 / 0.061 = 0.57$  
Popular: $0.29 \times 0.65 \times 0.39 \times 0.3 = 0.022 / 0.061 = 0.36$  
Sports: $0.19 \times 0.33 \times 0.37 \times 0.24 = 0.00557 / 0.061 = 0.09$

**Item #2:**

Grades: $0.52 \times 0.47 \times 0.25 \times 0.23 = 0.014 / 0.037 = 0.38$  
Popular: $0.29 \times 0.35 \times 0.22 \times 0.35 = 0.0078 / 0.037 = 0.21$  
**Sports: $0.19 \times 0.67 \times 0.28 \times 0.47 = 0.01675 / 0.037 = .45**

Problem 2: Plotting and Regression Using R

Write R code that reads the Football data and creates a scatterplot as follows:

1. One point on the plot for each game played
2. X-axis is Prediction
3. Y-axis is actual outcome (HomeScore minus AwayScore)
4. Simple linear regression line is shown
5. Point colors: yellow if the home team won, blue for a tie, green if the away team won
Answer:

```r
F <- read.csv("~/Football.csv")
attach(F)
for (i in 1:nrow(F)) {
  if (F[i,'HomeScore'] > F[i,'AwayScore']) F[i,'Outcome'] <- 1;
  if (F[i,'HomeScore'] == F[i,'AwayScore']) F[i,'Outcome'] <- 2;
  if (F[i,'HomeScore'] < F[i,'AwayScore']) F[i,'Outcome'] <- 3;
}
reg = lm(HomeScore-AwayScore ~ Prediction)
plot(Prediction, HomeScore-AwayScore,
  col=c("yellow","blue","green")[F$Outcome], pch=16)
abline(reg)
```

**Problem 3: kNN Classification Using R**

Write R code that creates a classifier for the Schoolkids data using R's k-nearest-neighbors (knn) function, taking into account more features than the three that you used in Problem 1. We've created two new data files, SchoolkidsTrain.csv and SchoolkidsTest.csv: SchoolkidsTest contains 20 rows extracted from the original Schoolkids data, and SchoolkidsTrain contains the rest. In both of the data files, the School column has been removed, and Goal is shifted to the last (9th) column.

Once you have your code running, experiment with different values of k. In addition, try using different subsets of the features for prediction instead of all eight features. How high can you get the accuracy? In the program you submit, make sure to use the setting for k and the set of features that give the highest accuracy you were able to find.

**Answer: (0.6 accuracy)**

```r
install.packages("class")
library(class)
S1 <- read.csv("~/SchoolkidsTrain.csv")
S2 <- read.csv("~/SchoolkidsTest.csv")
S1 <- transform(S1,Gender=as.numeric(Gender),Type=as.numeric(Type))
S2 <- transform(S2,Gender=as.numeric(Gender),Type=as.numeric(Type))
train <- S1[,c(2,4,5,6,7,8)]
test <- S2[,c(2,4,5,6,7,8)]
trainlabels <- S1[,9]
pred <- knn(train, test, trainlabels, k=60)
umtests <- nrow(test)
umcorrect <- 0
for (i in 1:numtests)
  { message("predicted:",pred[i]," actual:",S2[i,9]);
    if (pred[i] == S2[i,9]) numcorrect <- numcorrect+1 }
cat("Fraction correct: ", numcorrect/numtests)
```
Problem 4: Clustering Using R

(Football data) Write R code that clusters the Football data based on two features, HomeScore and AwayScore, and creates a plot with HomeScore on the x-axis, AwayScore on the y-axis, and point colors showing the clusters.

- Use only the games played in the first two weeks of 1998, i.e., only the first 30 rows of the Football data. To reference rows 1-30 in a dataframe D use D[1:30].

Using your program, experiment with different numbers of clusters from 2 to 8. Choose the number you like best and use it in the program that you submit; put a comment at the top of your code briefly justifying your choice.

Answer: (Four clusters gives nice separation into sensible groups without overfitting)

```R
F <- read.csv("~/Football.csv")
attach(F)
clus <- kmeans(F[1:30,c('HomeScore','AwayScore')], 4)
Fclus <- data.frame(F[1:30], clus$cluster)
with(Fclus, plot(HomeScore, AwayScore, col=clus$cluster, pch=16))
```

Problem 5: Sampling

Suppose you need to reduce the size of the football data through sampling. You decide to try several methods (note not all samples are the same size):

1. Random selection with 10% probability, i.e., each game has a 10% chance of being included in the sample
2. Every 10th game in the dataset, i.e., games 10, 20, 30, 40, etc.
3. Games played in “representative” weeks 4, 9, and 14
4. Games played in “representative” year 1999
5. Games involving “representative” teams Dallas, Denver, and Detroit (as home or away)

Write R code that creates the five samples, then prints a few simple statistics to compare the samples against the original data:

- Mean HomeScore
- Mean AwayScore
- Mean Prediction
- Mean error in Prediction (absolute difference between prediction and actual outcome)

The output of your program should consist of a total of 24 numbers: for each of the four statistics, you should give the result of that statistic on the original full dataset and on each of the five samples. You should structure the output so it’s easy for you (and the TA!) to compare the statistics across the different samples.
Just based on eyeballing the results, are any samples clearly better than others? Which one(s) do you like best? Put your answers in a comment at the top of your code.

**Answer:** (Overall One-Year seems to do best overall, not surprising since it's the largest sample, although interestingly Three-Teams does pretty well too)

```r
F <- read.csv("~/Football.csv")
attach(F)
F1 <- F[runif(nrow(F)) < 0.1,]
F2 <- F[as.integer(rownames(F)) %% 10 == 0,]
F3 <- F[Week == 4 | Week == 9 | Week == 14,]
F4 <- F[Year == 1999,]
F5 <- F[Home == 'Dallas' | Home == 'Denver' | Home == 'Detroit' |
        Away == 'Dallas' | Away == 'Denver' | Away == 'Detroit',]

message("Average Home Score")
m <- mean(F$HomeScore)
message(" Full Data: ", m)
m <- mean(F1$HomeScore)
message(" Random 10%: ", m)
m <- mean(F2$HomeScore)
message(" Every 10th: ", m)
m <- mean(F3$HomeScore)
message(" Three weeks: ", m)
m <- mean(F4$HomeScore)
message(" One year: ", m)
m <- mean(F5$HomeScore)
message(" Three teams: ", m)

message("Average Away Score")
m <- mean(F$AwayScore)
message(" Full Data: ", m)
m <- mean(F1$AwayScore)
message(" Random 10%: ", m)
m <- mean(F2$AwayScore)
message(" Every 10th: ", m)
m <- mean(F3$AwayScore)
message(" Three weeks: ", m)
m <- mean(F4$AwayScore)
message(" One year: ", m)
m <- mean(F5$AwayScore)
message(" Three teams: ", m)

message("Average Prediction")
m <- mean(F$Prediction)
message(" Full Data: ", m)
```
m <- mean(F1$Prediction)
message(" Random 10\%: ", m)
m <- mean(F2$Prediction)
message(" Every 10th: ", m)
m <- mean(F3$Prediction)
message(" Three weeks: ", m)
m <- mean(F4$Prediction)
message(" One year: ", m)
m <- mean(F5$Prediction)
message(" Three teams: ", m)

message("Average Prediction Error")
m <- mean(abs(F$Prediction - (F$HomeScore - F$AwayScore)))
message(" Full Data: ", m)
m <- mean(abs(F1$Prediction - (F1$HomeScore - F1$AwayScore)))
message(" Random 10\%: ", m)
m <- mean(abs(F2$Prediction - (F2$HomeScore - F2$AwayScore)))
message(" Every 10th: ", m)
m <- mean(abs(F3$Prediction - (F3$HomeScore - F3$AwayScore)))
message(" Three weeks: ", m)
m <- mean(abs(F4$Prediction - (F4$HomeScore - F4$AwayScore)))
message(" One year: ", m)
m <- mean(abs(F5$Prediction - (F5$HomeScore - F5$AwayScore)))
message(" Three teams: ", m)