Ranking Applications for Nursery Schools

by Viviane Adohouannon, Kate Alexander, Juan Arangote, Dian Azbel, Igor Baranov

Abstract Nursery Database was derived from a hierarchical decision model originally developed to rank applications for nursery schools.Out task was to develop a classification model to develop a reliable recomendation algorithm to predict is a specific student is sutable to be addmitted to a nursing school.

Introduction

This section may contain a figure such as Figure 1.

Figure 1: The logo of R.

Background

Introductory section which may include references in parentheses (R Core Team, 2012), or cite a reference such as R Core Team (2012) in the text.

Nursery Data is described as ...Sed ut dui dui. Vestibulum vel velit at mauris auctor gravida condimentum porttitor metus. Integer tempus nunc ac sem pharetra volutpat. Fusce vitae eleifend leo. Ut vel tempor nibh. Proin eget fermentum leo. Mauris pharetra vitae sem eget dictum. Sed at neque vitae metus lobortis luctus. Morbi sapien diam, vulputate sed diam vitae, pharetra accumsan mauris. Curabitur pretium nulla turpis, porta porta odio tempor vitae. In vehicula volutpat dui et consequat. Nam id lorem molestie sapien ultrices elementum. In vehicula metus elit, nec rhoncus est efficitur et. Proin ex tellus, vestibulum a eros at, maximus euismod justo.

Source: http://archive.ics.uci.edu/ml/datasets/Nursery

```
| class values

not_recom, recommend, very_recom, priority, spec_prior

| attributes

parents:     usual, pretentious, great_pret.
has_nurs:    proper, less_proper, improper, critical, very_crit.
form:        complete, completed, incomplete, foster.
children:    1, 2, 3, more.
housing:     convenient, less_conv, critical.
finance:     convenient, inconv.
social:      nonprob, slightly_prob, problematic.
health:      recommended, priority, not_recom.
```

Objective and Hypothesis

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Assessment of Situation

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae.

Plan

Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod:

- Cras sit amet lacus luctus massa lobortis sagittis.
- Curabitur convallis nisi non fringilla mattis.
- Etiam auctor massa nec orci hendrerit convallis.
- Nunc et tortor eu nisl gravida tempor a a lorem.
- Morbi sit amet sem posuere, aliquam ex ut, lacinia nulla.

Data understanding

Harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod.

Preparation

	X	parents	has_nurs	form	children	housing	finance	social	health	class
1	1	usual	proper	complete	1	convenient	convenient	nonprob	recommended	recommend
2	2	usual	proper	complete	1	convenient	convenient	nonprob	priority	priority
3	3	usual	proper	complete	1	convenient	convenient	nonprob	not_recom	not_recom
4	4	usual	proper	complete	1	convenient	convenient	slightly_prob	recommended	recommend
5	5	usual	proper	complete	1	convenient	convenient	slightly_prob	priority	priority
6	6	usual	proper	complete	1	convenient	convenient	slightly_prob	not_recom	not_recom
7	7	usual	proper	complete	1	convenient	convenient	problematic	recommended	priority
8	8	usual	proper	complete	1	convenient	convenient	problematic	priority	priority
9	9	usual	proper	complete	1	convenient	convenient	problematic	not_recom	not_recom
10	10	usual	proper	complete	1	convenient	inconv	nonprob	recommended	very_recom

Table 1: `Nursery Data Dataset (head)`

Summary of Nursery Data set is extracted by the folloving 'summary' function, the results are presented in Table 2 and Table 3.

We can conclude that: consectetur adipiscing elit. Aenean magna urna, sodales vel blandit sed, condimentum sit amet ante. Phasellus pulvinar ullamcorper porttitor. Cras vitae ipsum in magna condimentum malesuada ut at massa. Duis quis quam faucibus, euismod lacus sit amet, scelerisque odio.

	X	parents	has_nurs	form	children
X	Min. : 1	great_pret :4320	critical :2592	complete :3240	1 :3240
X.1	1st Qu.: 3241	pretentious:4320	improper :2592	completed :3240	2 :3240
X.2	Median : 6480	usual :4320	less_proper:2592	foster :3240	3 :3240
X.3	Mean : 6480		proper :2592	incomplete:3240	more:3240
X.4	3rd Qu.: 9720		very_crit :2592		
X.5	Max. :12960				

Table 2: `Summary of Nursery Data Dataset, columns 1-5`

Distribution of Class attribute presented in Figure 2. Morbi volutpat augue vitae porta lobortis. Integer sit amet neque vel risus aliquam scelerisque et eget est. Cras maximus ex nec pharetra dictum. Vivamus vehicula ante sodales massa rhoncus, et blandit tortor interdum. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus

	housing	finance	social	health
X	convenient:4320	convenient:6480	nonprob :4320	not_recom :4320
X.1	critical :4320	inconv :6480	problematic :4320	priority :4320
X.2	less_conv :4320		slightly_prob:4320	recommended:4320
X.3				
X.4				
X.5				

Table 3: Summary of Nursery Data Dataset, columns 6-9

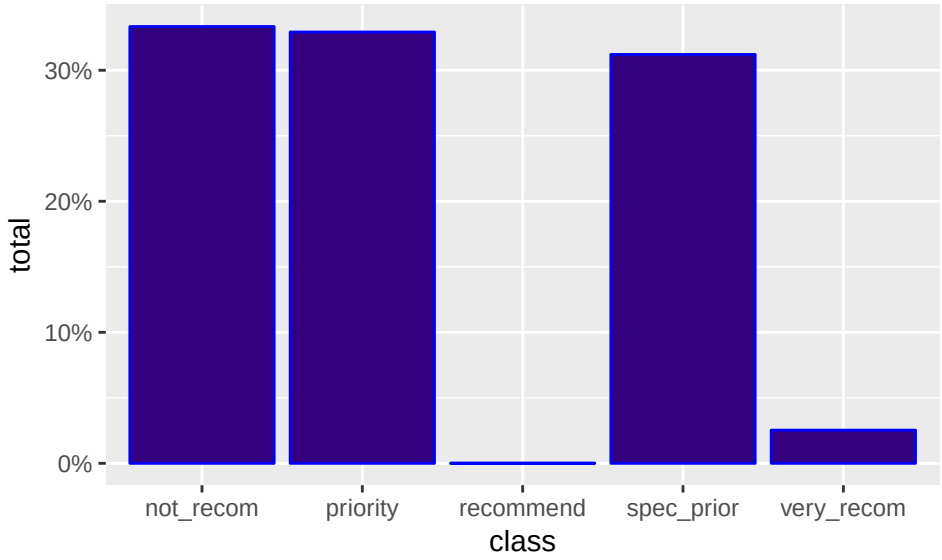

Figure 2: Distribution of Class attribute

vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
prop.table((table(nursery_data$class)))

#>
#>   not_recom    priority   recommend  spec_prior  very_recom
#> 0.333333333 0.329166667 0.000154321 0.312037037 0.025308642

ggplot(nursery_data, aes(x=as.factor(class) )) +
  geom_bar(aes(y= (..count..)/sum(..count..)),color="blue",fill=rgb(0.2,0,0.5)) +
  theme(legend.position = "none") +
  scale_y_continuous(labels=scales::percent) +
  labs(x = "class",y="total")
```

The following code (Figure 3) renders distribution of class attribute depending on parents attribute. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
ggplot(nursery_data,aes(x=factor(parents),fill=factor(class)))+
  geom_bar(position="dodge")
```

The following code (Figure 4) renders distribution of class attribute depending on 'health' attribute. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis

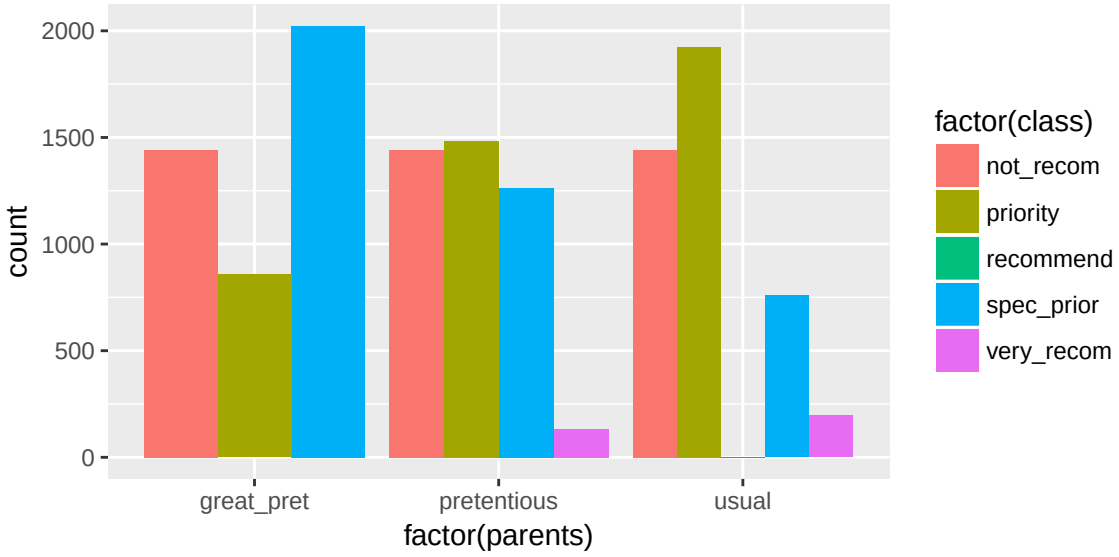

Figure 3: Distribution of Class attribute depending on parents

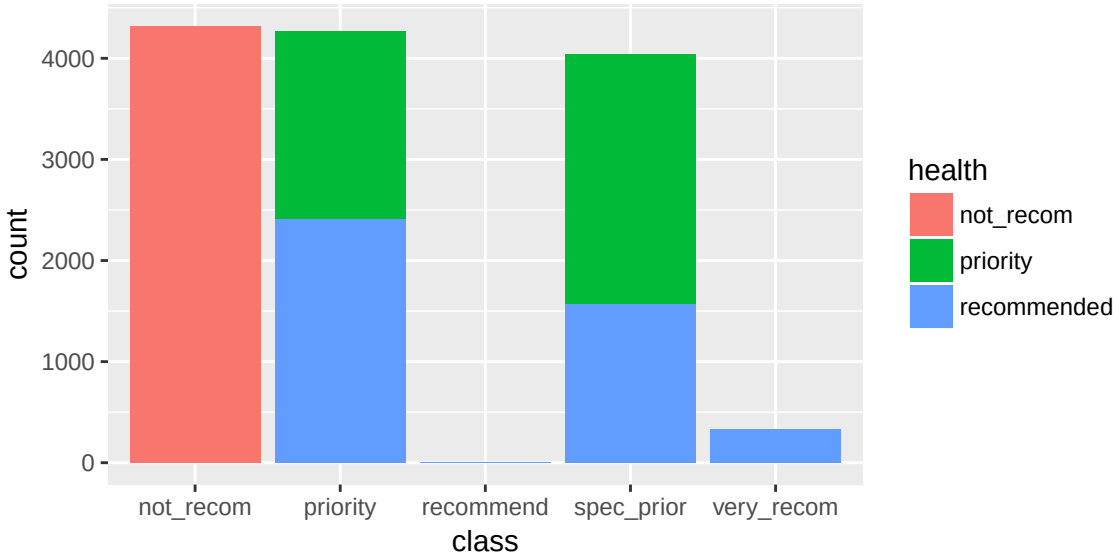

Figure 4: Distribution of Class attribute depending on health

tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
nursery_data$health <- as.factor(nursery_data$health)
ggplot(data = nursery_data, mapping = aes(x = class, fill = health)) +
  geom_bar()
```

The following code (Figure 5) renders distribution of class attribute depending on ''social' attribute. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
ggplot(nursery_data, aes(social, fill=class)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), alpha=0.9) +
  facet_wrap(~parents) +
  scale_fill_brewer(palette = "Dark2", direction = -1) +
  scale_y_continuous(labels=scales:::percent, breaks=seq(0,0.4,0.05)) +
```

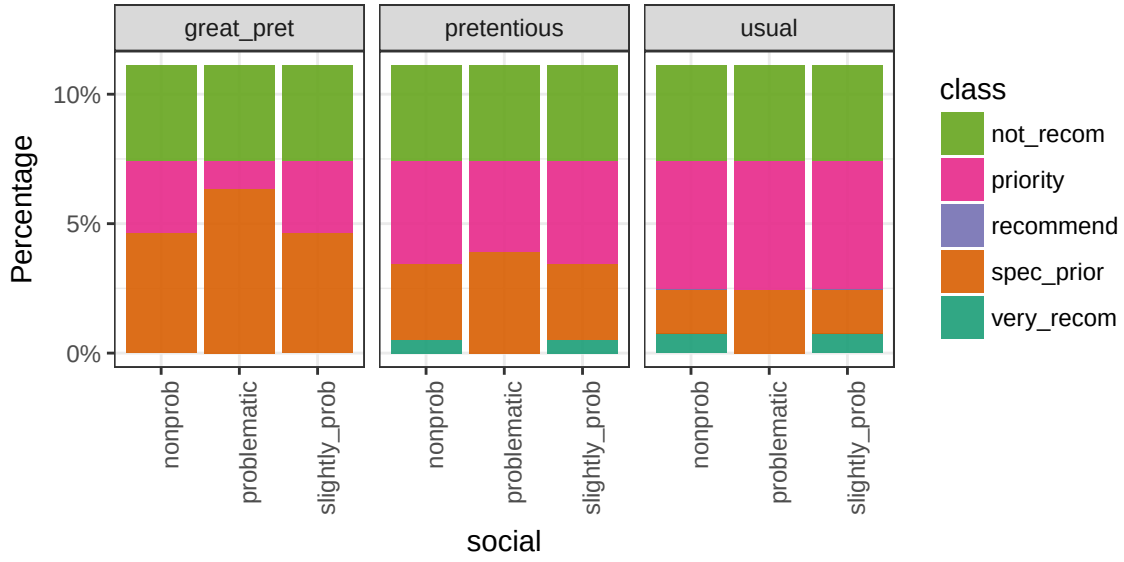

Figure 5: Distribution of Class attribute depending on income

```
  ylab("Percentage") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

The following code (Figure 5) renders distribution of 'class' attribute depending on 'nursery' attribute. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
ggplot(nursery_data, aes(form, fill=class)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), alpha=0.9) +
  facet_wrap(~has_nurs) +
  scale_fill_brewer(palette = "Dark2", direction = -1) +
  scale_y_continuous(labels=scales:::percent, breaks=seq(0,0.4,0.05)) +
  ylab("Percentage") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

The conclusion is: aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

Modeling

Splitting the dataset into train and test

We are splitting the dataset in such a way, that train and test sets would have similar distribution of the 'class' attribute.

```
train.rows<- createDataPartition(y= nursery_data$class, p=0.9, list = FALSE)
train.data<- nursery_data[train.rows,]
prop.table((table(train.data$class)))
```

```
#>
#>    not_recom     priority    recommend   spec_prior   very_recom
#> 0.3332761872 0.3291616664 0.0001714384 0.3120178296 0.0253728785
```

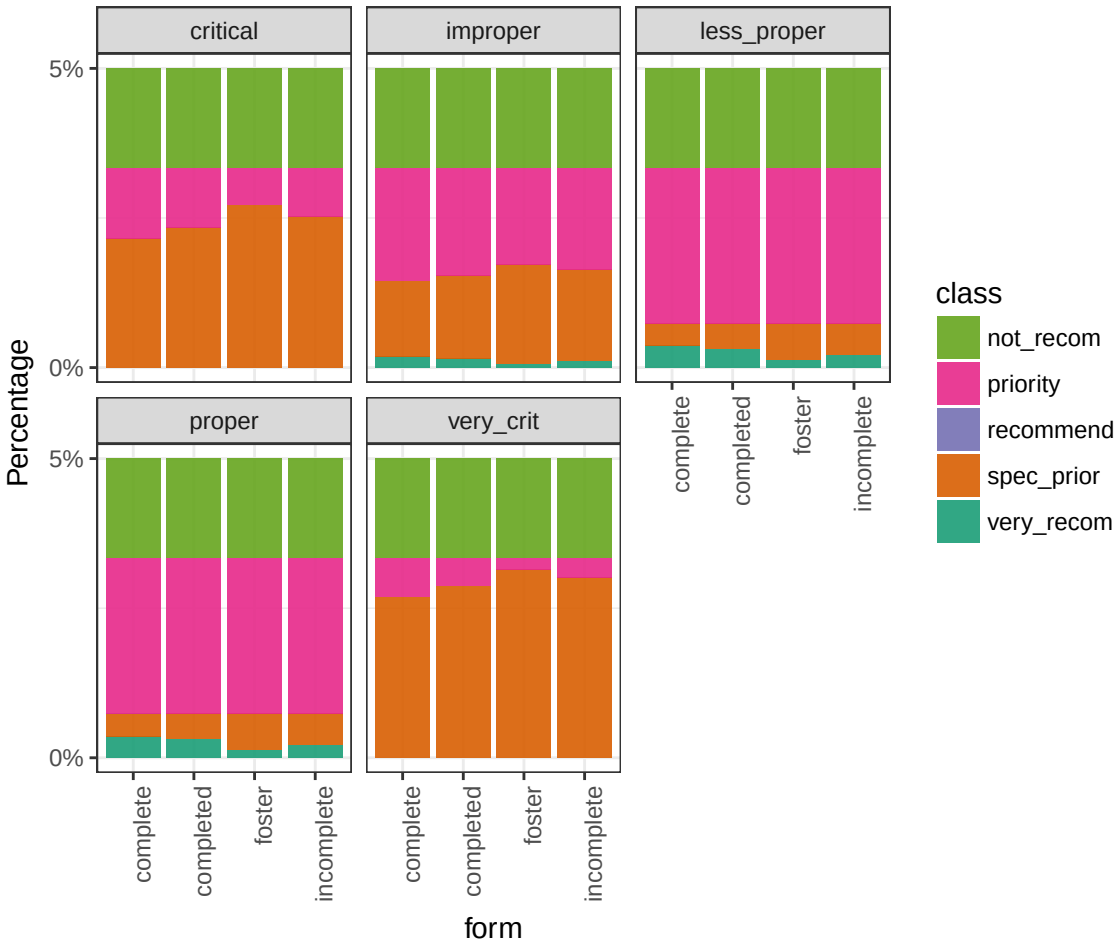

Figure 6: Distribution of Class attribute depending on nursery

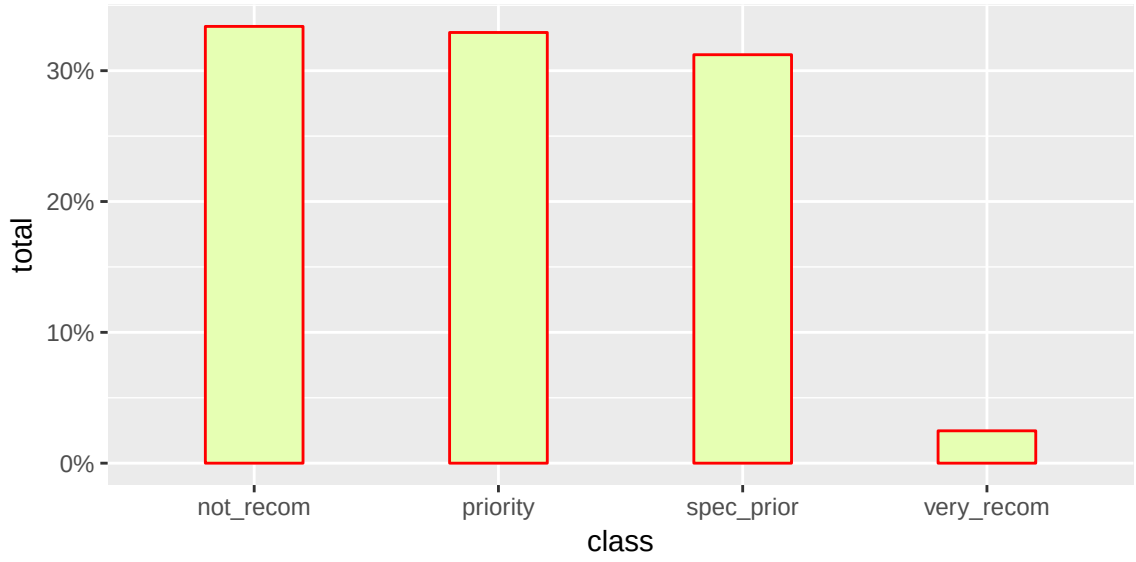

Figure 7: Distribution of Class attribute depending on nursery

```
test.data<- nursery_data[-train.rows,]
prop.table((table(test.data$class)))

#>
#>  not_recom   priority  recommend spec_prior very_recom
#> 0.33384853 0.32921175 0.00000000 0.31221020 0.02472952
```

The following code (Figure 7) renders distribution of 'class' attribute depending on 'nursery' attribute. Pellentesque aliquam ligula eu justo porttitor, non fringilla erat vehicula. Pellentesque et dolor non nunc aliquet euismod. Vivamus vel malesuada lorem. Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
ggplot(test.data, aes(x=as.factor(class))) +
  geom_bar(aes(y = (..count..)/sum(..count..)),width=0.4,
  color="red", fill=rgb(0.9,1,0.7) )+theme(legend.position = "none") +
  labs(x = "class",y="total")+scale_y_continuous(labels=scales:::percent)
```

Decision Tree model fit

As a first step we will train a Desigion Tree model. This model is known to be computationally fast, but not very precise. We will use all default parameters and all attributes of thetrain dataset. Resuling tree is presented in Figure 8. It shows that Fusce eget mauris a nulla sollicitudin eleifend eu auctor ligula. Sed nec dictum lorem. Vestibulum bibendum ultrices lorem, id fermentum felis tincidunt eu. Curabitur ipsum justo, dictum id finibus vitae, pulvinar non tortor. Curabitur vel mi a urna gravida commodo vitae vel libero. Maecenas imperdiet sed diam eget viverra.

```
fitdt <- rpart(as.factor(class)~., method="class", data=train.data)
fancyRpartPlot(fitdt, main = "", sub = "")
```

Decision Tree model evaluation

```
dtPrediction <- predict(fitdt, test.data, type = "class")

confMat <- table(dtPrediction,test.data$class)
confMat

#>
#> dtPrediction not_recom priority recommend spec_prior very_recom
```

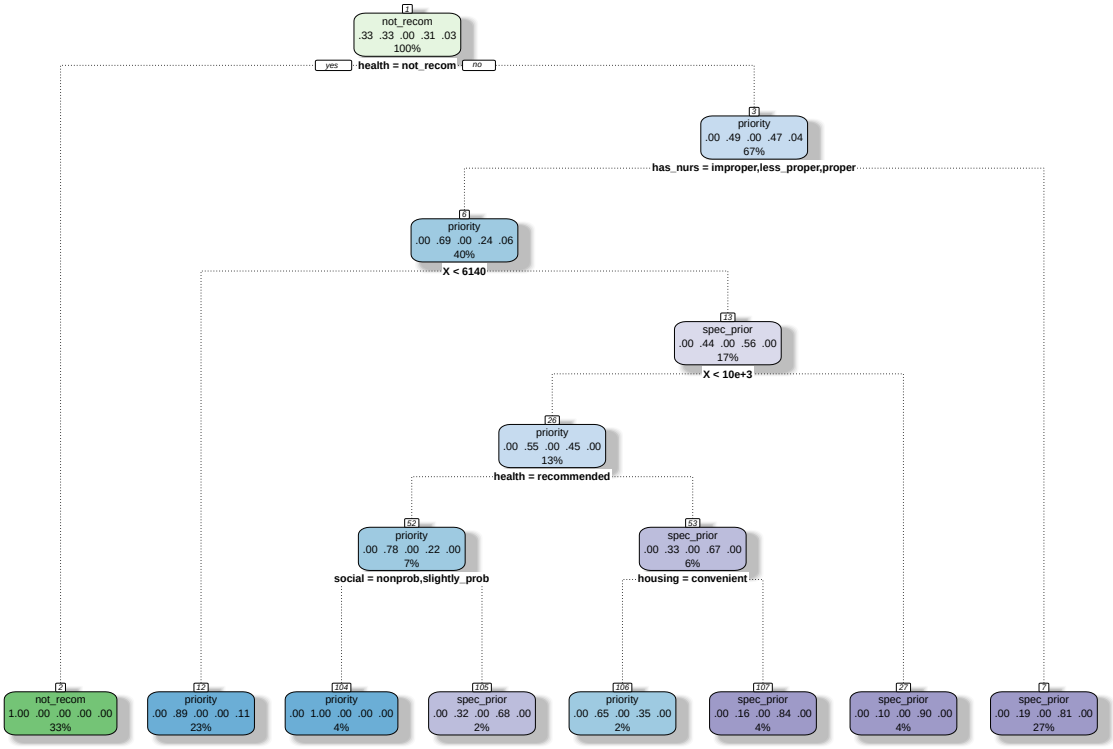

Figure 8: Decision tree diagram

```
#>   not_recom        432        0          0          0          0
#>   priority           0      325          0         11         32
#>   recommend          0        0          0          0          0
#>   spec_prior         0      101          0        393          0
#>   very_recom         0        0          0          0          0
```

```
accuracy <- sum(diag(confMat))/sum(confMat)
cat(sprintf("\nAccuracy=%f", accuracy))
```

```
#>
#> Accuracy=0.888717
```

Random Forest model fit

```
library(randomForest)
fitRF1 <- randomForest(as.factor(class)~.,
                       data=train.data,
                       importance=TRUE,
                       ntree=1000)
```

Importance of the dataset attributes for the prediction of the 'class' attribute shown in Figure 9. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat.

```
varImpPlot(fitRF1, main="")
```

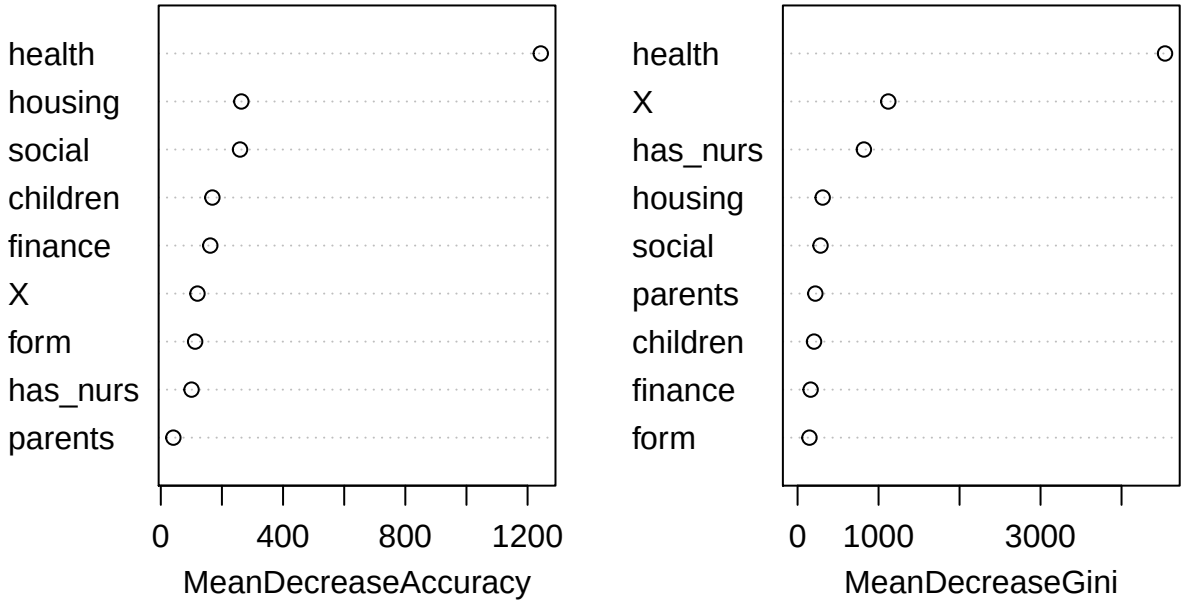

Figure 9: Importance of the dataset attributes for the prediction of the 'class' attribute

Random Forest model prediction and evaluation

```
PredictionRF1 <- predict(fitRF1, test.data)
head(PredictionRF1)

#>         7         9        33        48        53        60
#>  priority not_recom not_recom not_recom  priority not_recom
#> Levels: not_recom priority recommend spec_prior very_recom

confMat <- table(PredictionRF1,test.data$class)
confMat

#>
#> PredictionRF1 not_recom priority recommend spec_prior very_recom
#>    not_recom        432        0         0          0          0
#>    priority           0      425         0          0          0
#>    recommend          0        0         0          0          0
#>    spec_prior         0        1         0        404          0
#>    very_recom         0        0         0          0         32

accuracy <- sum(diag(confMat))/sum(confMat)
cat(sprintf("\nAccuracy=%f", accuracy))

#>
#> Accuracy=0.999227
```

Dicsussion

Nor again is there anyone who loves or pursues or desires to obtain pain of itself, because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to

obtain some advantage from it? But who has any right to find fault with a man who chooses to enjoy a pleasure that has no annoying consequences, or one who avoids a pain that produces no resultant pleasure?"

Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

This file is only a basic article template. For full details of *The R Journal* style and information on how to prepare your article for submission, see the Instructions for Authors.

Bibliography

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. URL http://www.R-project.org/. ISBN 3-900051-07-0. [p1]

Viviane Adohouannon
York University School of Continious Studies

https://learn.continue.yorku.ca/user/view.php?id=21444

Kate Alexander
York University School of Continious Studies

https://learn.continue.yorku.ca/user/view.php?id=21524

Juan Arangote
York University School of Continious Studies

https://learn.continue.yorku.ca/user/view.php?id=21472

Dian Azbel
York University School of Continious Studies

https://learn.continue.yorku.ca/user/view.php?id=20687

Igor Baranov
York University School of Continious Studies

https://learn.continue.yorku.ca/user/profile.php?id=21219