Solution 1:

a) The spam data is a binary classification task where the aim is to classify an email as spam or no-spam.

```
library(mlr3)
library(mlr3learners)
library(mlr3filters)

## Error in library(mlr3filters):  there is no package called 'mlr3filters'

tsk("spam")

## <TaskClassif:spam> (4601 x 58)
## * Target: type
## * Properties: twoclass
## * Features (57):
##   - dbl (57): address, addresses, all, business, capitalAve,
##     capitalLong, capitalTotal, charDollar, charExclamation, charHash,
##     charRoundbracket, charSemicolon, charSquarebracket, conference,
##     credit, cs, data, direct, edu, email, font, free, george, hp, hpl,
##     internet, lab, labs, mail, make, meeting, money, num000, num1999,
##     num3d, num415, num650, num85, num857, order, original, our, over,
##     parts, people, pm, project, re, receive, remove, report, table,
##     technology, telnet, will, you, your
```
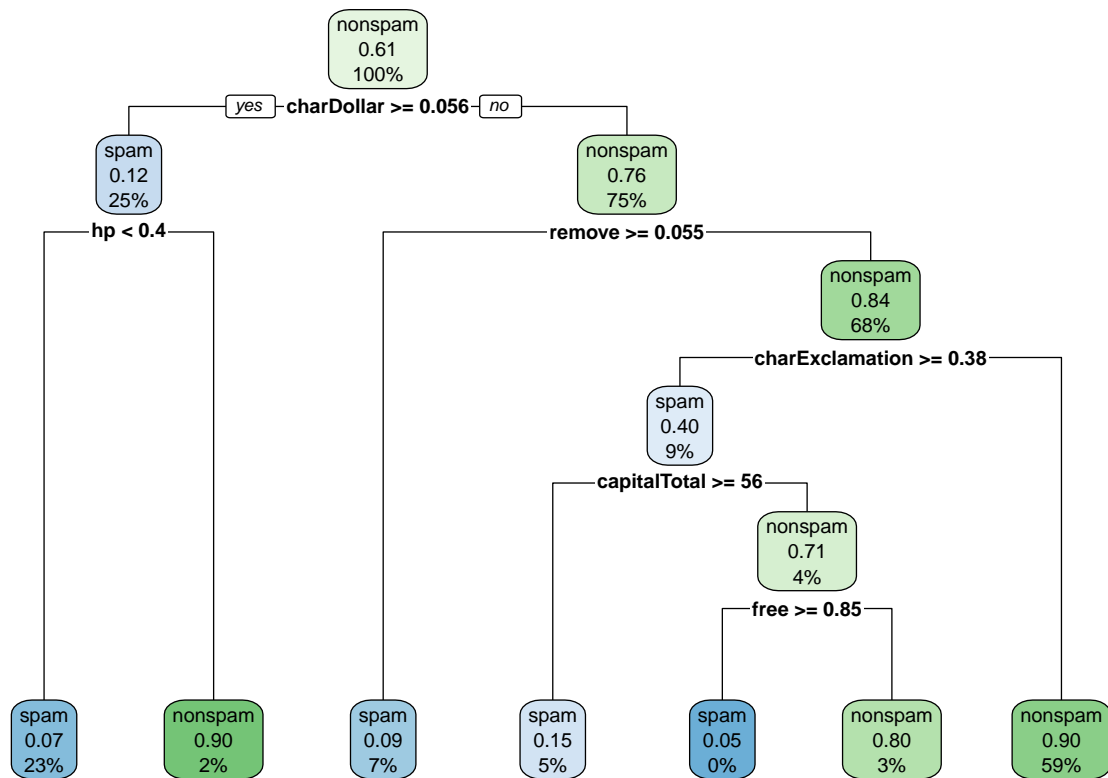
b)
```
library(rpart.plot)

## Loading required package:  rpart

task_spam <- tsk("spam")

learner <- lrn("classif.rpart")
learner$train(task_spam)

rpart.plot(learner$model, roundint=FALSE)
```
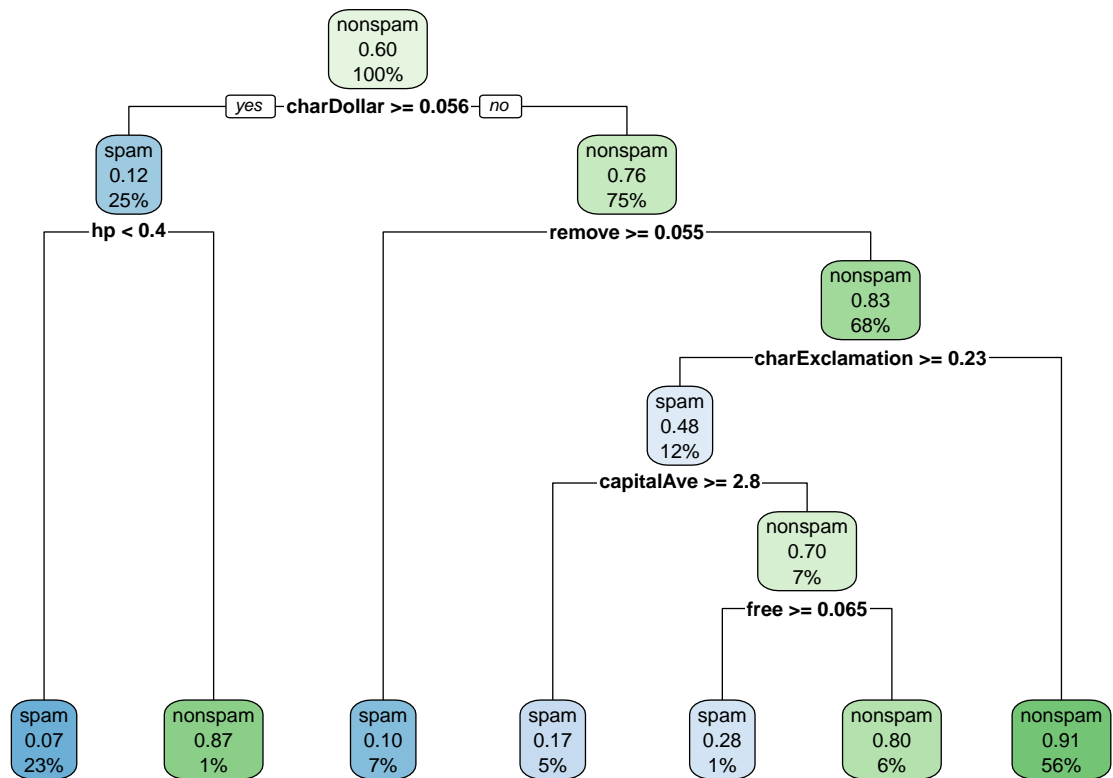
```r
set.seed(42)

subset1 <- sample.int(task_spam$nrow, size = 0.8 * task_spam$nrow)
subset2 <- sample.int(task_spam$nrow, size = 0.8 * task_spam$nrow)

learner$train(task_spam, row_ids = subset1)
rpart.plot(learner$model, roundint=FALSE)
```
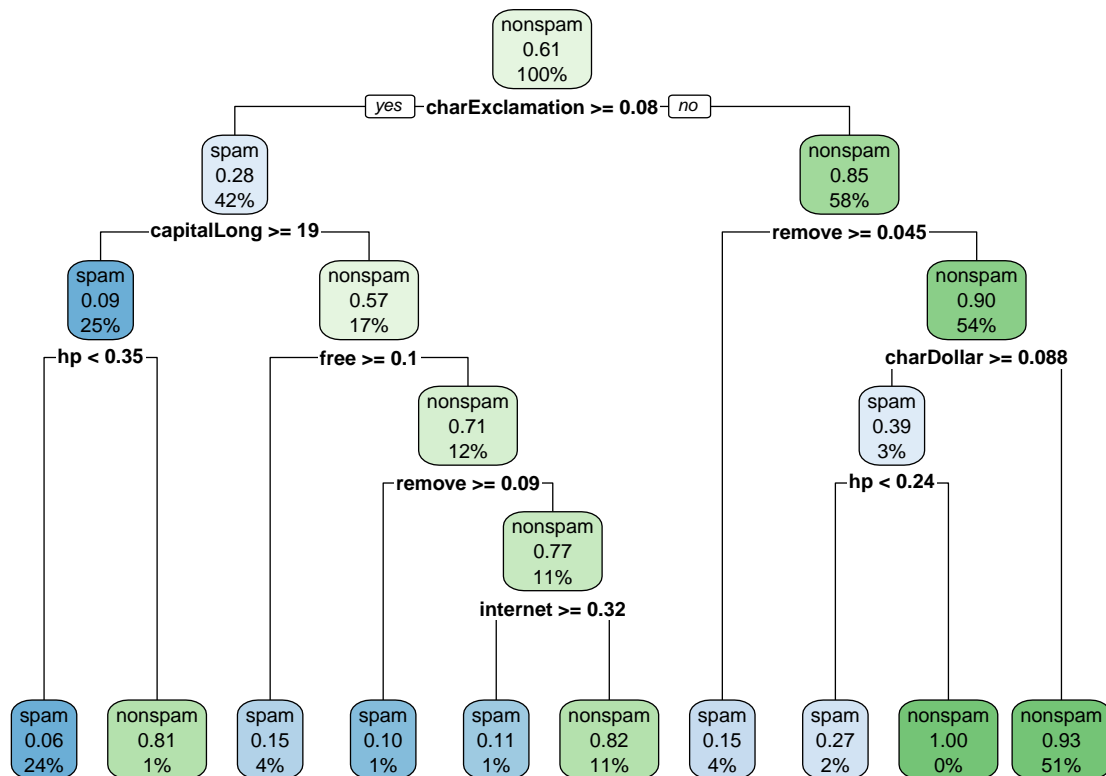
```
learner$train(task_spam, row_ids = subset2)
rpart.plot(learner$model, roundint=FALSE)
```

nonspam
0.61
100%

yes — **charExclamation >= 0.08** — no

spam
0.28
42%

nonspam
0.85
58%

capitalLong >= 19

spam
0.09
25%

nonspam
0.57
17%

remove >= 0.045

nonspam
0.90
54%

hp < 0.35

free >= 0.1

nonspam
0.71
12%

charDollar >= 0.088

spam
0.39
3%

remove >= 0.09

nonspam
0.77
11%

hp < 0.24

internet >= 0.32

spam
0.06
24%

nonspam
0.81
1%

spam
0.15
4%

spam
0.10
1%

spam
0.11
1%

nonspam
0.82
11%

spam
0.15
4%

spam
0.27
2%

nonspam
1.00
0%

nonspam
0.93
51%

Observation: Trees with different sample find different split points and variables, leading to different trees!

c)
```r
learner <- lrn("classif.ranger", "oob.error" = TRUE)
learner$train(tsk("spam"))

model <- learner$model

model$prediction.error

## [1] 0.04542491
```

d) Variable importance in general measures the contributions of features to a model. One way of computing the variable importance of the j-th variable is based on permutations of the OOB observations of the j-th variable, which measures the mean deacrease of the predictive accuracy induced by this permutation. To determine the n variables with the biggest influence on the prediction quality, one can choose the n variables with the highest variable importance based on permutations of the OOB, e.g. for $n = 5$:

```r
learner <- lrn("classif.ranger", importance = "permutation", "oob.error" = TRUE)
filter <- flt("importance", learner = learner)

## Error in flt("importance", learner = learner):  could not find function "flt"

filter$calculate(tsk("spam"))

## Error in filter$calculate:  object of type 'closure' is not subsettable

head(as.data.table(filter), 5)

## Error in as.data.frame.default(x, ...):  cannot coerce class '"function"' to a
data.frame
```

Solution 2:
See R code