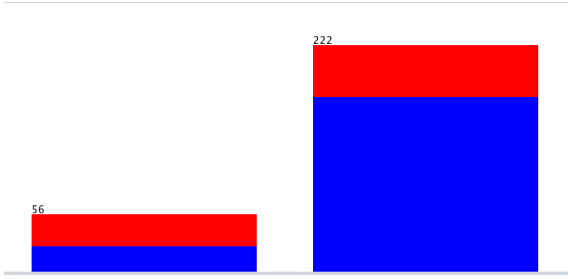
(a) Class is presented as a binary attribute with two possible values: no-recurrence-events and recurrence-events

(b) age - ordered
menopause - categoric
tumor-size - ordered
inv-nodes - ordered
node-caps - binary
deg-malig - ordered (degree of malignancy)
breast - binary
breast-quad - categoric
irradiat - binary

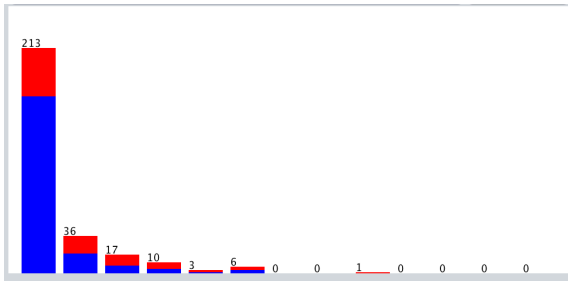
(c) The best statistical distribution for age appears to be a normal distribution, due to the symmetrical bell-shaped curve for the age graph. According to this, we can estimate the parameters for such a normal distribution:

mean = 51.143
variance = 102.3776
standard deviation = 10.118

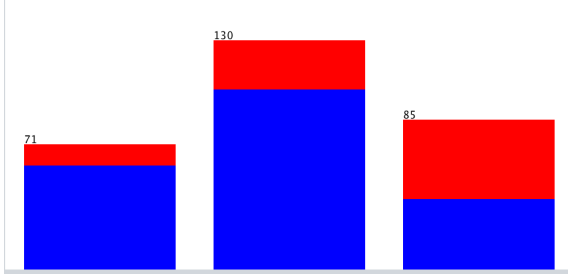
(d) node-caps, inv-nodes, deg-malig, irradiat, tumor-size



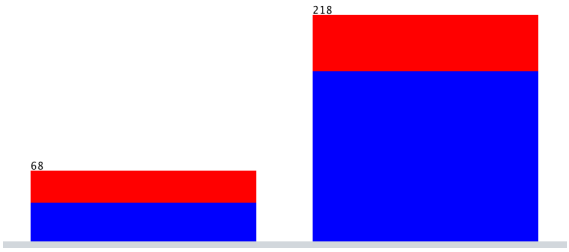
According to the graphic, for node-caps we can see that if node-caps = no then the probability of class = no-recurrence-events is higher than the probability of class = recurrence-events.



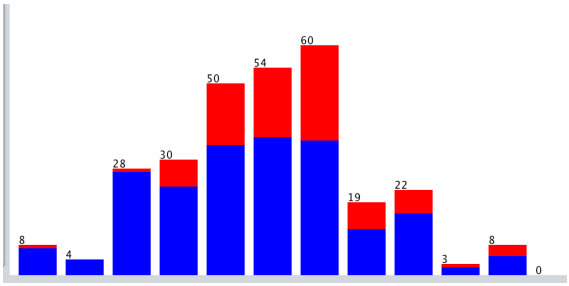
For inv-nodes, if inv-nodes = 0-2 then the probability of class = no-recurrence-events is higher than he probability of class = recurrence-events, however, if if inv-nodes = 24-26, then class = recurrence-events is more expected.



We can also notice the dependencies for deg-malig. The more the degree of malignancy, the more class = recurrence-events is predictable.



For irradiat we can notice that if irradiat = no then the probability of class = no-recurrence-events is higher than he probability of class = recurrence-events.



There is also a dependence between tumor-size and class label. For example, if tumor-size = 5-9 or 10-14 then class = no-recurrence-events is more predictable compared to the case when tumor-size = 30-34.

Exercise 2

(a) Zeror, accuracy = 65.9794 %

a	b	<-- classified as
64	0	a = no-recurrence-events
33	0	b = recurrence-events

ZeroR is the simplest classification method which relies on the target and ignores all predictors. According to the confusion matrix and detailed accuracy by class, The ZeroR algorithm predicts only “no-recurrence-events” value for all instances as it is the majority class(201 out of 286 samples), and achieves an accuracy of 65.9794%.

(b) Naive Bayes, accuracy = 71.134 %

(c) IBk Euclidean-Distance

k = 1 72.1649 %
k = 2 69.0722 %
k = 3 70.1031 %
k = 4 73.1959 %
k = 5 73.1959 %
k = 6 73.1959 %
k = 7 73.1959 %
k = 8 72.1649 %
k = 9 70.1031 %
k = 10 70.1031 %

Values for the best accuracy: k = 4, k = 5, k = 6, k = 7

(d) IBk Manhattan-Distance

k = 1 72.1649 %
k = 2 69.0722 %
k = 3 70.1031 %
k = 4 73.1959 %
k = 5 73.1959 %
k = 6 73.1959 %
k = 7 73.1959 %

k = 8 72.1649 %
k = 9 70.1031 %
k = 10 70.1031 %

Values for the best accuracy: k = 4, k = 5, k = 6, k = 7

The results for both types of distances are the same. The reason for this is the fact, K-Nearest Neighbours algorithm produces the nearest neighbour approach. Using different methods of finding distance changes the distance, but the nearest neighbour still stays the same. As a result, the classifier shows the same accuracies. Moreover, Euclidean Distance and Manhattan Distance are usually appropriate when there are continuous numerical variables in the data. In our case, all variables are discrete.

(e) J48, accuracy = 68.0412 %

if node-caps == yes:
  if deg-malig == 1: recurrence-events
  if deg-malig == 2: no-recurrence-events
  if deg-malig == 3: recurrence-events
else if node-caps == no: no-recurrence-events

Number of Leaves : 4
Size of the tree : 6

The main aim of J48 is to get the smallest tree. The heuristic is measuring the amount of information contained in a set of data. It gives the idea of importance of an attribute in a dataset. In this particular case, node “node-caps” is considered to contain much more information, so it has been selected as the first split criteria.
According to the tree, if node-caps = no then the majority of samples fall in “no-recurrence-events”. Once again “deg-malig” is the attribute which contains much more information and that is the reason why the second leaf of the tree starts from the attribute “deg-malig”.

Overfitting occurs when we achieve a good fit of the model on the training data, while it does not generalize well on new, unseen data. In other words, the model learned patterns specific to the training data, which are irrelevant in other data.
We can identify overfitting by looking at the accuracies of the classifier before and after splitting the data. Accuracy with using training set is 75.8741%, however, splitting the data provides less accuracy (68.0412%). So the model is overfitting the data.

(f) SVM, accuracy = 70.1031%

The most heavily weighted attributes are:
-0.5056 * (normalized) inv-nodes=0-2
-0.5631 * (normalized) node-caps=no
+0.6353 * (normalized) deg-malig=3

The weights represent the hyperplane, that separates the classes as best as possible. The sign before weigh value means the class label (- for no-recurrence-events, + for recurrence-events). In contrast, J48 uses another approach, information gain, to decide, in each tree node, which variable fits better in terms of target variable prediction. So there is a difference in defined most heavily weighted attributes: it is deg-malig for SVM and node-caps for J48.

Exercise 3

The lowest false positive rate for no-occurrence-events class is achieved using the NaiveBayes Classifier and is equal to 0.515.
The highest precision for the no-recurrence-events class is also achieved using the NaiveBayes Classifier and is equal to 0.757.

The false positive rate is calculated as the ratio between the number of negative events (recurrence-events) wrongly categorized as positive (false positives) and the total number of actual negative events (recurrence-events). In this particular case, fp = 17 / (17+16) = 0.515. The classifier wrongly predicts 51.5% of people to not have breast cancer while they have it; as a result, they will not get treatment. This is quite poor for practical application as many people will not get the treatment they need. According to the fact, that it is the lowest rate, we can't accept this rate for the application of predicting recurrence of breast cancer.

Precision attempts to find the proportion of positive identifications that are actually correct. It is calculated as the ratio between the number true positive events (no-recurrence-events) and the total number of events classified as positive(no-recurrence-events). In this case, precision = 53/(53+17) = 0.757. It means that 75.7% of people, who were predicted to not have cancer, do not actually have it. It is the highest value among all the classifiers. However, it is high enough to be applied in predicting.

Exercise 4

(a) The number of examples for each value for a given attribute is identical. This suggests that the data was intentionally chosen to represent each combination of attributes equally, with the exception of target class.

(b) weka.classifiers.functions.supportVector.NormalizedPolyKernel, accuracy = 96.4286 %

In the linearly separable case, SVM is trying to find the hyperplane that maximizes the margin, with the condition that classes are classified correctly. But in reality, datasets are rarely linearly separable. In our case, the data seems to be highly separated. For example, there are no class = good or class = vgood for buying = vhigh. What Kernel Trick does is it utilizes existing features, applies some transformations, and creates new features. Those new features are the key for SVM to find the nonlinear decision boundary. Support vector machine with a polynomial kernel can generate a non-linear decision boundary using polynomial features.

Exercise 5

(a) Single attribute values automatically ensure that the vehicle is unacceptable:
persons = 2
safety = low

(b) J48, accuracy = 94.2177 %

SVM, accuracy = 94.3878 %

In my opinion, J48 is easy to interpret due to the opportunity to visualise the tree and predict outcomes by traversing it from the root node to the leaf node. It is relatively quicker to get the result as it is based on calculation of information gain compared to SVM where it is solving convex optimization case. Also, SVM are less interpretable. It is non-trivial to explain why the classification works this way.