Assignment 3

Please follow the following rules when working on this assignment:

- This assignment is mandatory for all students except data science students.
- Work on this assignment in teams of three students
- You can use your own computer to work on this assignment. If your team wants to use a VM that has already WEKA installed, write an email to Dennis Tschechlov (Dennis.Tschechlov@ipvs.uni-stuttgart.de).
- **Prepare a document** (PDF) as follows:
  - You may write this document in English or German.
  - The PDF has to include the team number, name, study programme and matriculation number for each team member.
  - Document name: A3-Txx-Vy.pdf (where "xx" is the team number and "y" the version).
  - Provide answers to questions, which are marked by *Qt.n: …* where "t" is the number of the task and "n" the number of the question for this task. For each Question, explain your answer in 2 – 3 sentences if not explicitly stated otherwise.
  - Provide screenshots of your results for certain steps. We mark explicitly when you should provide such a screenshot with *St.n*.
  - Your result document should not include answers or screenshots related to tasks 1.3, 2.3 and 3.3. Instead, you will present the results for these tasks in a live presentation (see below).

- **Submit your document** in Ilias no later than February 10, 10:00 am. This works as follows:

  - Go to 'Submit Assignments' and select Assignment 3.
  - One team member has to select 'Create Team' and afterwards 'Manage Team' to add your team members. Choose 'Add Users of Current Course' to add your team members.
  - Note that you can only select individual students and not directly the teams/groups we already have organized in Ilias. Anyway, the teams you are creating to submit your document have to match one of the available teams.
  - Finally click 'Hand in' to submit the prepared PDF.
  - Please make sure that you submit only one document per team and that you add all team members before submitting your document.

- Each team has to make an appointment to **present the results** of Task 1.3, 2.3 and 3.3 (15 to 30 minutes per team).

  - Each group member has to actively participate in this result presentation, i.e., each team member should present the results for one task. You should be prepared to present the results using Weka on your laptop or on the given VM via WebEx. You may also prepare slides for the presentation, but this is not mandatory. Note that we have only 30 minutes per team.
  - Result presentations are scheduled for February. Note that the result presentation can only take place once you have submitted the result document as described above.
  - Go to the following doodle to vote for your preferred dates for the result presentation. Vote as a team and vote for at least six time slots on at least three different days! We need your response by January 10!
    - https://doodle.com/poll/46uw6fcwy8mq4k33
- Contact Dennis Tschechlov (Dennis.Tschechlov@ipvs.uni-stuttgart.de) for any further questions or use the forum in Ilias.

Important**:** See document connect_vms.pdf to learn how to connect to your virtual machine providing these tools.

Introduction

In this assignment you will become familiar with the basic data mining techniques clustering, classification and association rules. You will apply them to datasets and get insights into the usage of these techniques. The goal is to apply the algorithms you learned in the lecture in practice on real-world datasets. To this end, you will apply different algorithms with several parameters to observe their effects. To apply the different techniques you will use the WEKA data mining tool, which implements several data mining algorithms and offers a GUI for exploring the data and visualizing the results.

Preliminaries: Getting Started with WEKA

You can download WEKA from here: https://waikato.github.io/weka-wiki/downloading_weka/

Download the appropriate version for your OS and proceed as described on the website. Make sure to use a stable version and not the experimental one.

Follow the setup assistant and afterwards, you should be able to start WEKA. On startup, the Weka GUI chooser should open. In this assignment, we will only use the "Explorer".
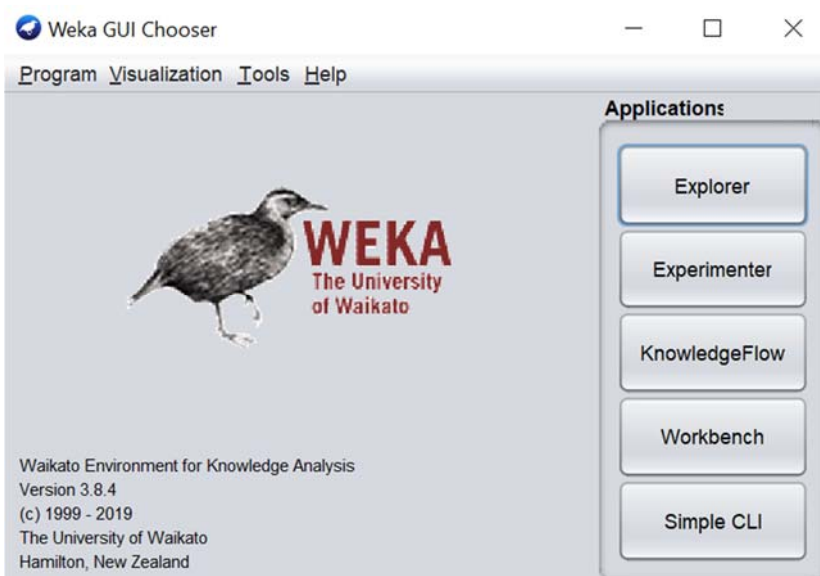
After clicking on Explorer, you should see the WEKA Explorer window, which has 6 Tabs:

- Preprocess: Here, you can select the dataset you want to process and can make further preprocessing steps. In this assignment, you will work with already preprocessed data. However, note that preprocessing data is mandatory in real-world applications and can be a very time-consuming task. All other tabs should be disabled now since you first have to select a dataset before you can proceed.

- Cluster, Classify and Associate: In these tabs you can apply the respective data mining technique.

- Select attributes: Here, you can apply methods that automatically choose the attributes to use for the data mining techniques, e.g., a Principal Component Analysis.

- Visualize: In this tab, you can visualize your dataset.
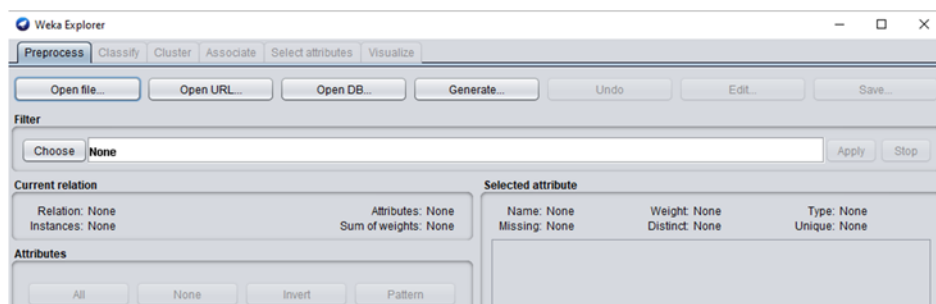
Next, we will select a dataset and visualize this dataset. To this end, click on Open file …. Then, go to the WEKA data directory. On Windows, this is by default *C:\Program Files\Weka-*

2

3-8-4\data and select the dataset *iris.arff*. You should now see a screen that describes the dataset:
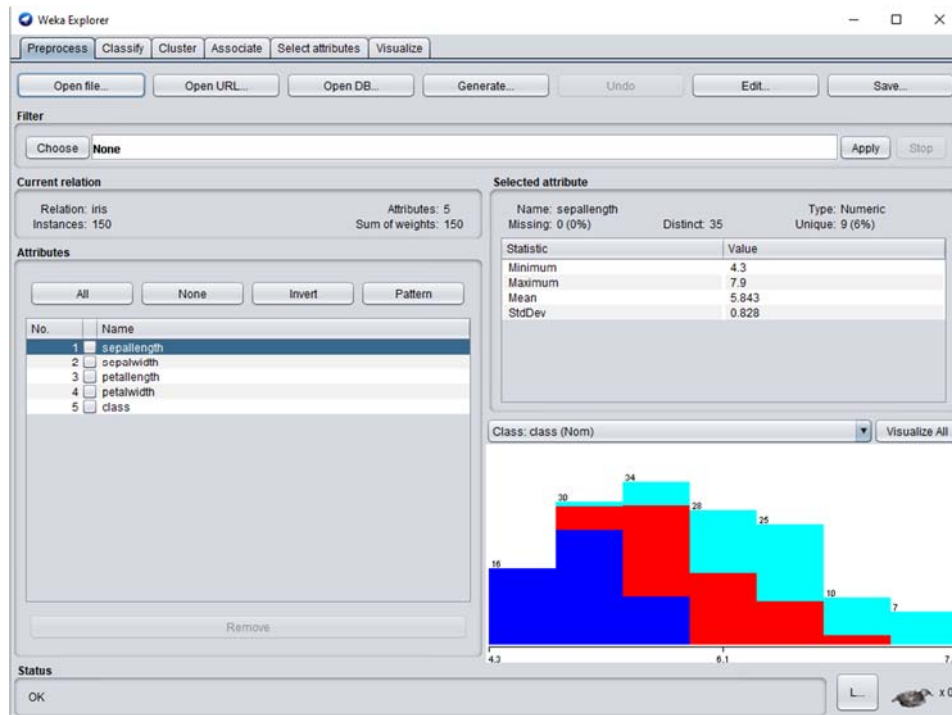
Afterwards, you can click on the Visualize tab to visualize the dataset. There you can see a pairplot that shows two attributes in a scatterplot. Play around with the PlotSize, PointSize and Jitter parameters to make the plots and the points larger. Do not forget to click on Update to make the changes visible.
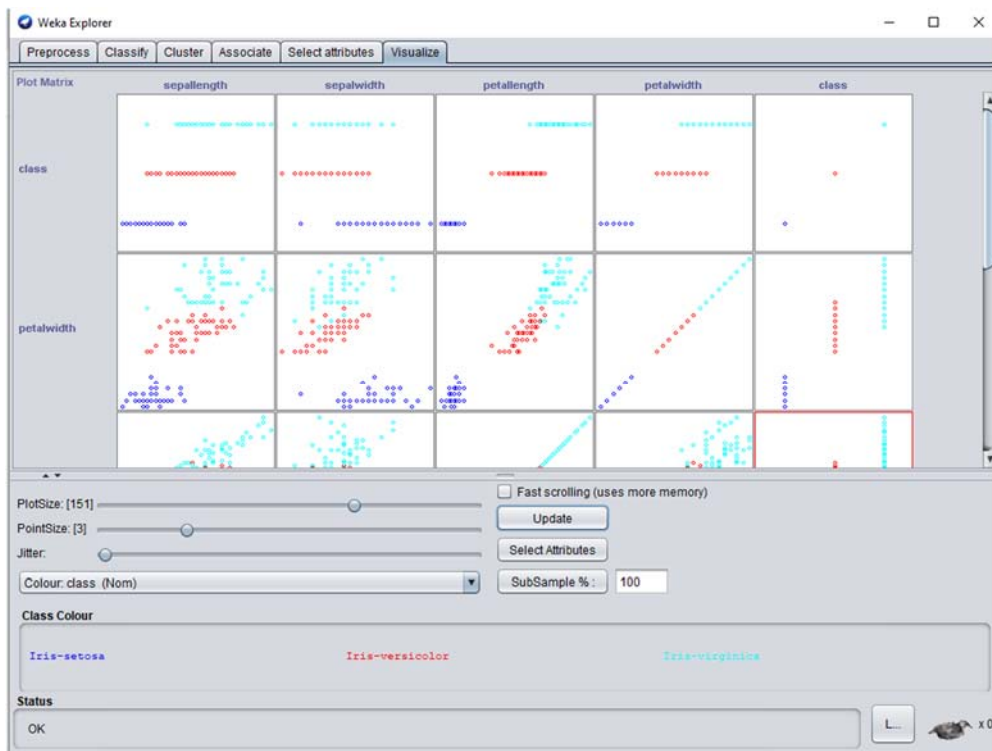
Task 1: Clustering

In this task, you will perform a clustering analysis. To this end, you will load the dataset (1.1), run a clustering algorithm (1.2) and explore different algorithms and parameters (1.3).

1.1 Import the dataset

- Click on Open file … in the Preprocess tab and load the dataset *small_gaussian_dataset.csv* that we provided with this assignment. Make sure to select CSV as file format.

- You should now see the dataset with its two attributes and statistics about the dataset.

- You can go to the Visualize tab and look at the visualization of the dataset.

   *S1.1*: **Provide a screenshot of the plots in the Visualize tab.**

    Q1.1: **When you look at the plots that plot *attr1* and *attr2* in one plot, what do you think how many clusters does this dataset have?**

1.2 Run KMeans on the imported dataset

- Click on tab Cluster.

- Click on Choose and select Simple KMeans.
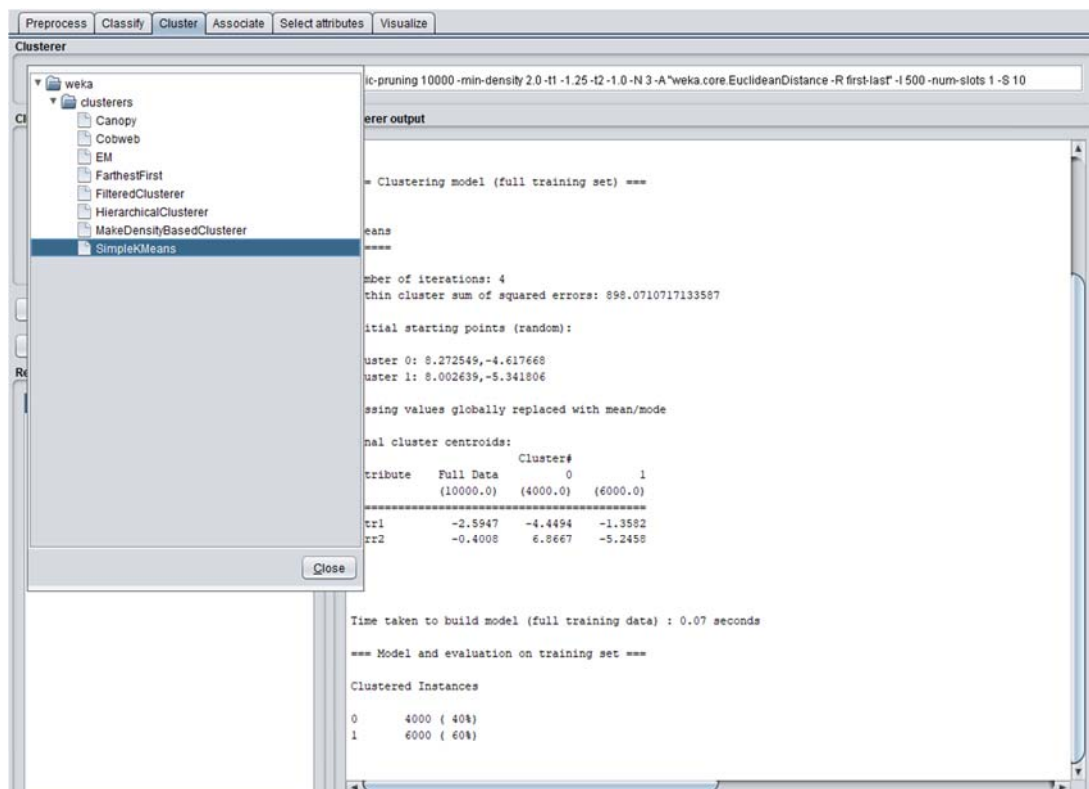

- Make sure that the checkbox Store clusters for Visualization is ticked.

- Click on Start.

- WEKA should now run KMeans on the dataset and should display an output on the console like the following:
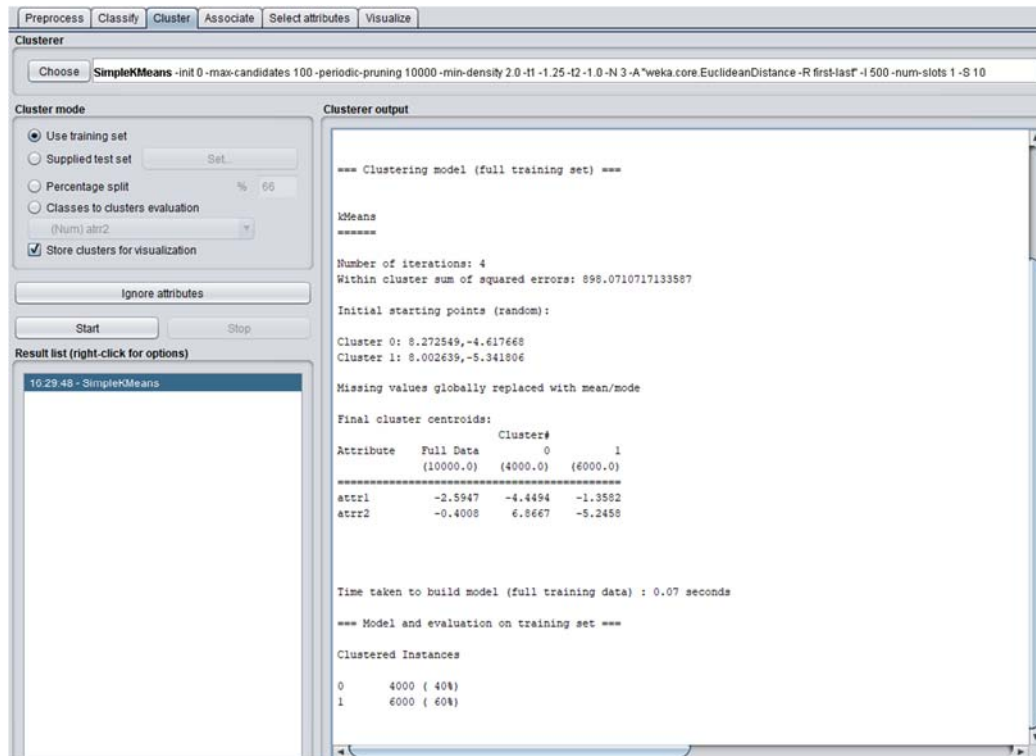
■ You can now right-click on the Result list on the left and select Visualize Cluster assignments. You should now see a visualization of how the data is partitioned into clusters. Make sure to select *attr1* on the x-axis and *attr2* on the y-axis.

S1.2: **Provide a screenshot of the visualized cluster assignments.**
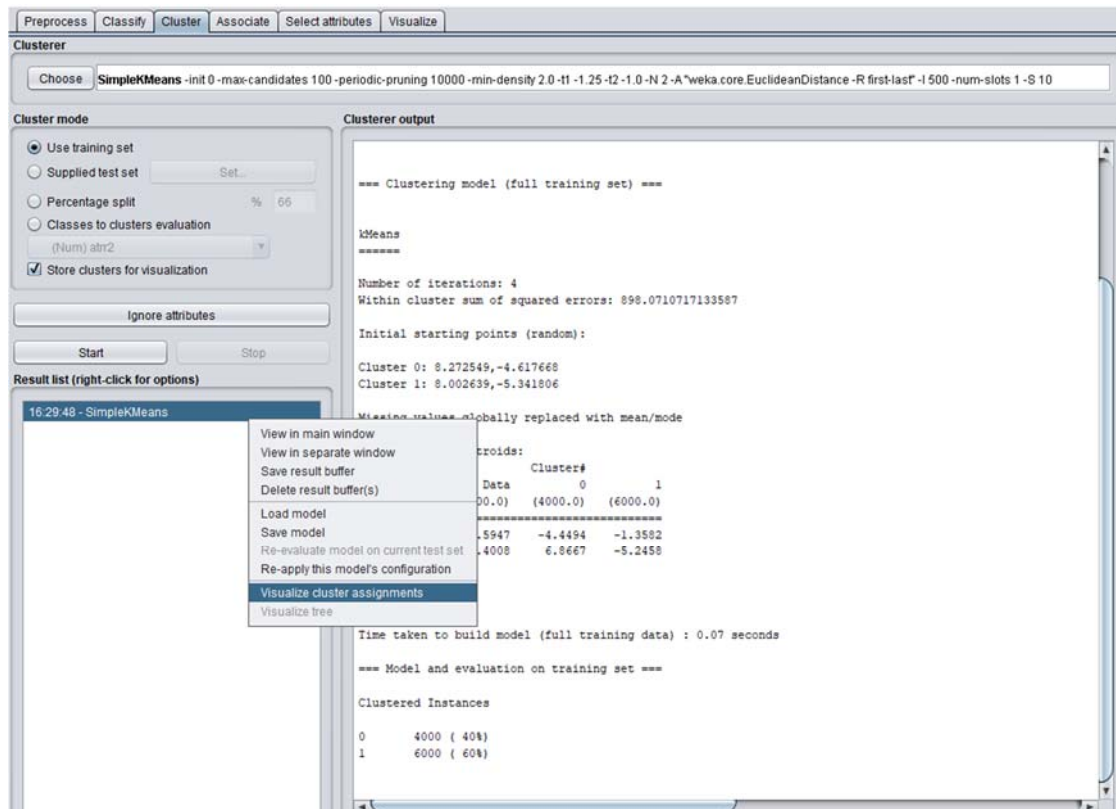
■ Repeat the previous steps of running the KMeans algorithm with different parameters. To this end, click next to choose on SimpleKMeans and then you should see a list of parameters.

■ Set NumClusters to 3, i.e., you execute KMeans with k=3 clusters. Leave the other parameters as default values like shown in the following screenshot.
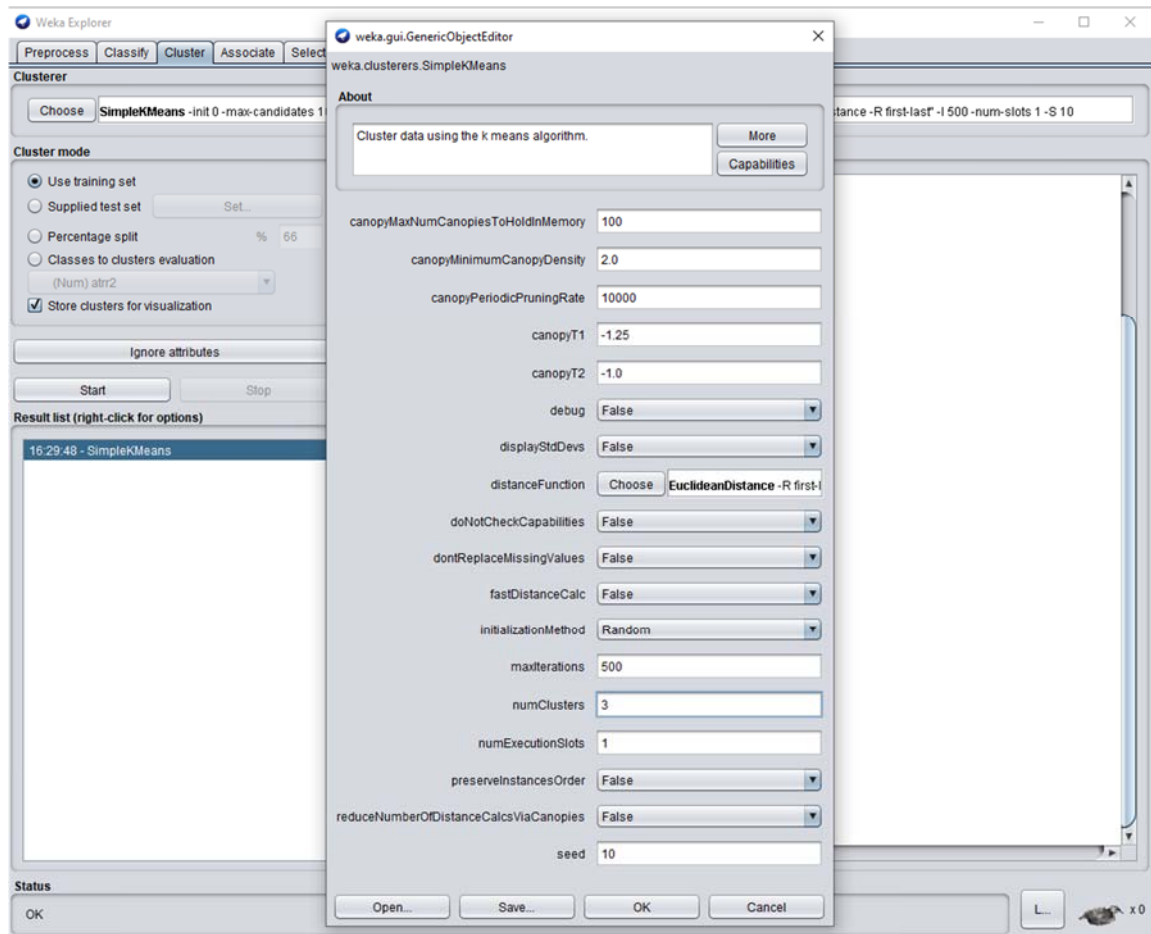
■ Run the algorithm again and visualize the results.

■ Repeat this procedure for k=4 and k=5.

S1.3: Provide a screenshot of the visualized cluster assignments for k=3, 4 and 5.

Q1.2: **How do the results differ? Does KMeans detect the actual clusters of the dataset?**

■ Run KMeans again with the number of clusters that you think are the actual number of clusters from Task 1.1, but this time choose k-means++ for the parameter InitializationMethod. Also, run the algorithm with seed=1 and seed=10.

S1.4: Provide a screenshot of the visualized cluster assignments for this parameter setting.

Q1.3: Do you get different results if you only change the seed parameter? If yes, Explain why this is the case.

Q1.4: **Does the algorithm find the actual clusters in the dataset? If yes, for which parameter configuration? You can make a screenshot to show the parameter configuration.** `seed:1, k:5`

Task 1.3 Explore Clustering Algorithms

In this task, we will use a different dataset, the *spiral* dataset, that has a very different structure compared to the dataset from the previous task:

■   First import the dataset *spiral.csv* as described before and visualize it to get familiar with the dataset. What do you think is the actual number of clusters of the dataset?

■   Run the SimpleKMeans algorithm on this dataset with at least three different values for the number of clusters. Use *k-means++* as InitializationMethod. What do you observe? Is KMeans able to detect the structures in the dataset?

■   Run at least two different clustering Algorithms that WEKA offers. Try to use algorithms from different families (density-based, partition-based, hierarchy-based …) of clustering algorithms. Your goal is to detect the shapes in the structure of the dataset.

■   Try three different parameter settings for each algorithm.

■   Run the same algorithms and parameter settings with the *small_gaussian_dataset.csv* (see Task 1.1). Do you observe a difference in the runtime of the different algorithms? You may consider the console output of WEKA to find information about the runtime.

What is your conclusion: Is it a good idea to always run KMeans as clustering algorithm and with the same parameter setting for all datasets? Explain your answer.

HierachyCluster

Task 2: Association Rule Discovery

In this task, you will perform an association rule discovery. To this end, you will load the dataset (2.1), run an algorithm to discover association rules (2.2) and explore different algorithms and parameters (2.3).

2.1 Import the dataset

■ Click on Open file … in the Preprocess tab and load the dataset *vote.arff* that we provided with this assignment. Make sure to select Arff as file format.

■ You should now see the dataset with its two attributes and statistics about the dataset.
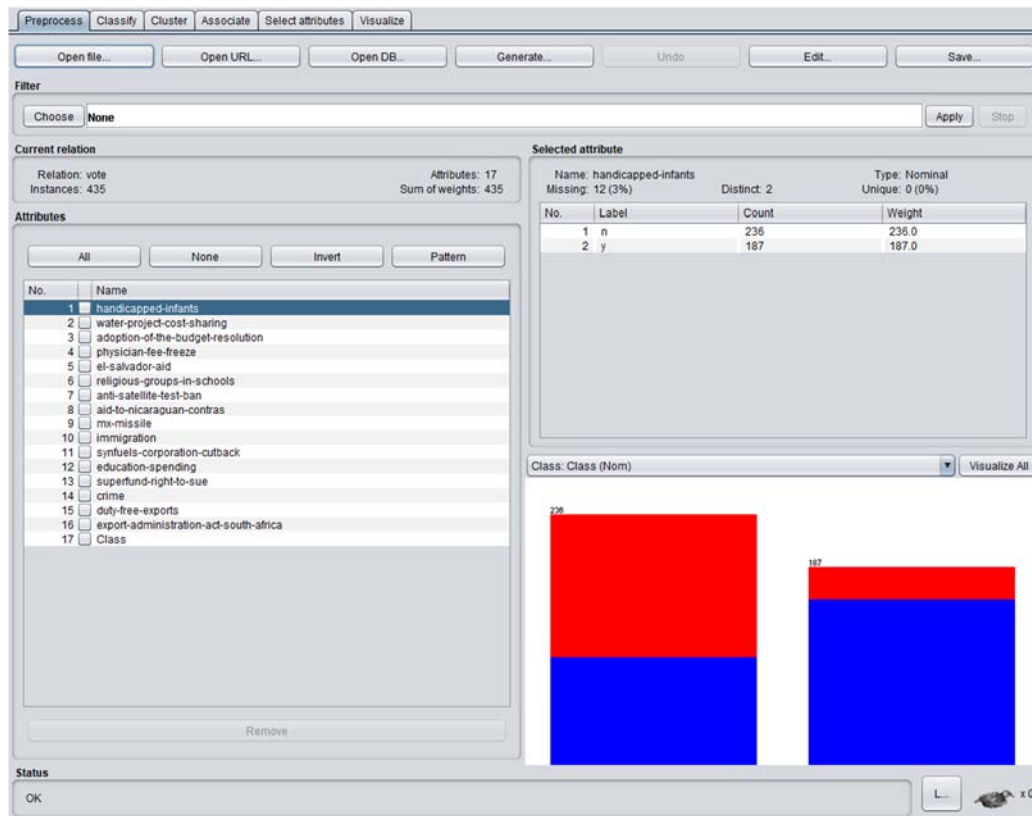
■ In the Attributes section, click on the attribute class to see more information about this attribute.

S2.1: Provide a screenshot of the result when you clicked on the class attribute (similar to the given screenshot but for the class attribute).

Democrate, Republic

Q2.1: What are the possible underline values for the class attribute and how often does each value occur? What do you think does this dataset describe (from just looking at the attributes and in particular on the class attribute)?

Q2.2: Could you also use this dataset for classification? Could you also use this dataset for the association rule discovery if you would remove the class attribute from the dataset? Yes, No

這邊要在想一下
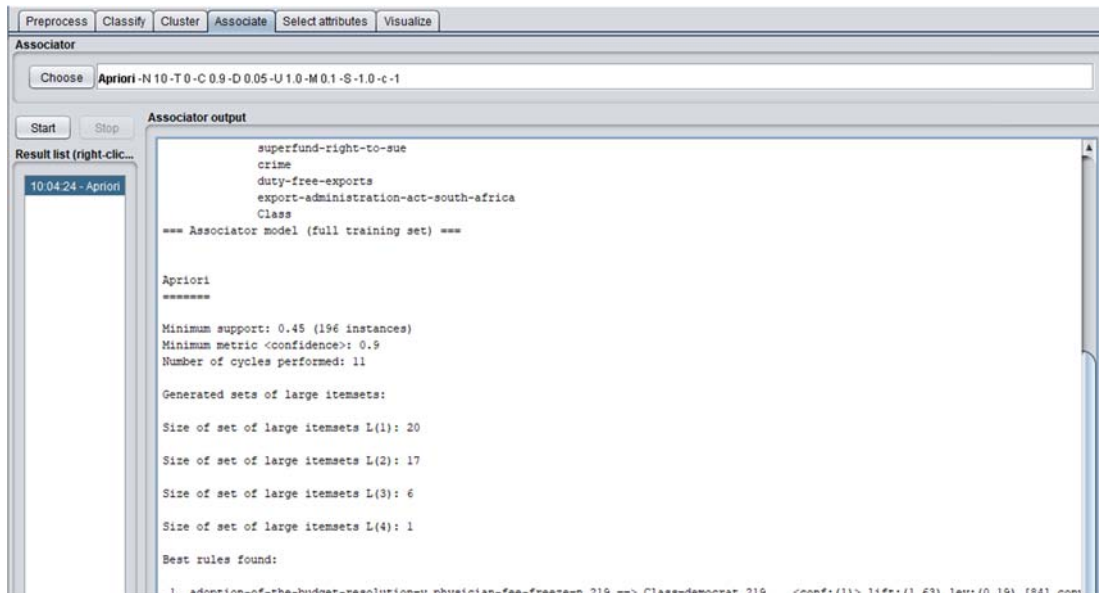
2.2 Run Apriori on the imported dataset

■ Click on tab Associate.

■ Click on Choose and select Apriori.
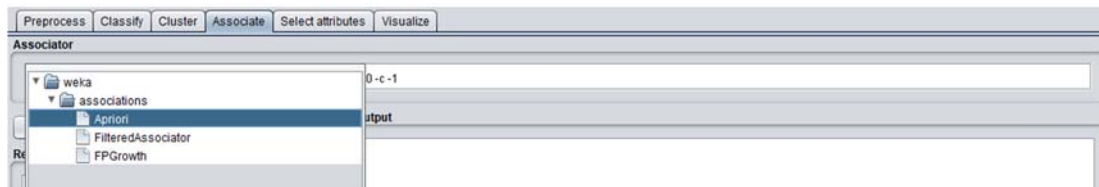
■ Click on the button Start. The output should look like the following:

■ The output shows the minimum <u>support</u>, the minimum <u>metric</u> (confidence per default) value, the number of cycles the algorithm performed and the itemsets together with the size of each. Below this, the output shows the <u>best rules</u> that are found by the algorithm. The left side of each rule shows for how many instances the premise of each rule occurred and on the right side for how many instances the implication hold. Moreover, the output shows the <u>confidence (conf.), lift</u>, etc. of the found rules.

S2.2: Provide a screenshot of the console output that shows the 10 best found rules.

Q2.3: For how many instances of the dataset did it occur that physician-fee-free was "no"? For how many did it occur that physician-fee-free is "no" <u>and</u> the class is "democrat"?

Q2.4: <u>List all rules that show under</u> which conditions people voted for a democrat. Consider only rules with a <u>confidence value of at least 0.99. What does the confidence express?</u>

5. physician-fee-freeze=n 247 ==> Class=democrat 245   <conf: (0.99)> lift:(1.62) lev:(0.21) [93] conv:(31.8)

245/435 / (247/435) * (267/435) =

2.3 Explore different algorithms and parameters

■ Next to Choose, click next to Apriori to see a list of the parameters. Set the parameter upperBoundMinSupport to the value 0.1, i.e., the same as the lowerBoundMinSupport.
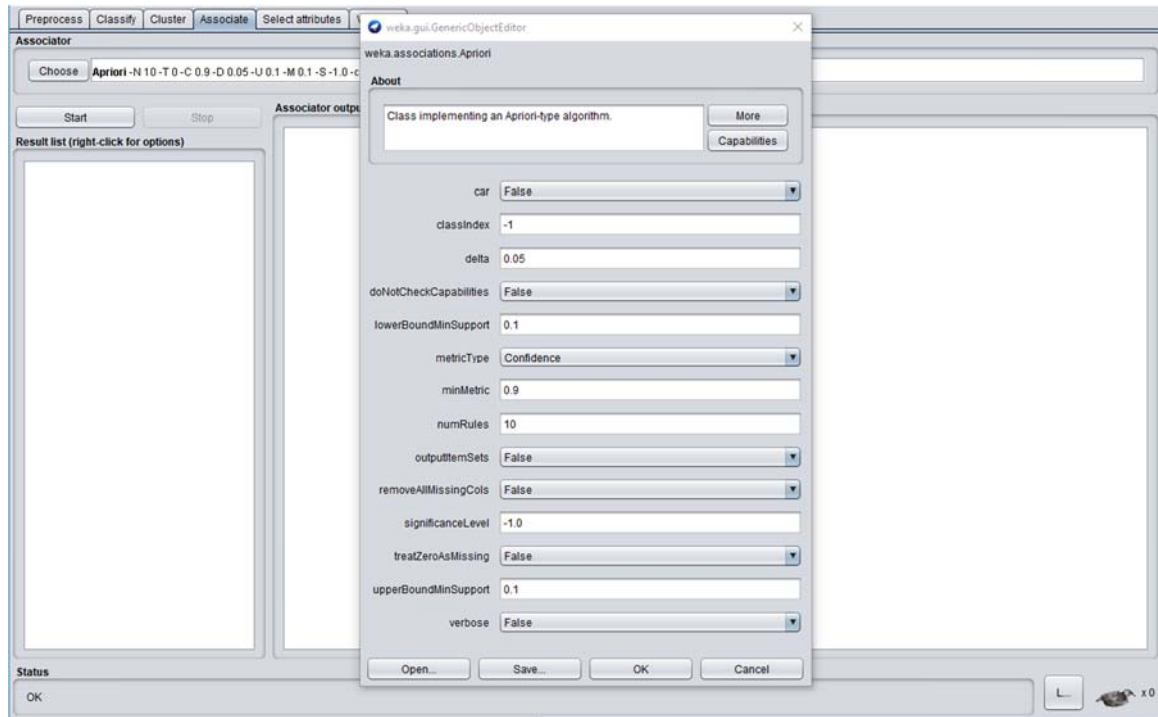
■ Click on OK and run the algorithm again by clicking on the Start button. What do you observe regarding the runtime? How often do the best found rules occur? What is the number of cycles performed in contrast to the previous results from Task 2.2?

■ Now, click on Choose and select the FPGrowth algorithm. FPGrowth is an algorithm for association rule discovery similar to the Apriori algorithm. You did not learn details about this algorithm in the lecture.

■ Here, also change the parameters by clicking next to FPGrowth and set the values of upperBoundMinSupport and lowerBoundMinSupport to 0.1.
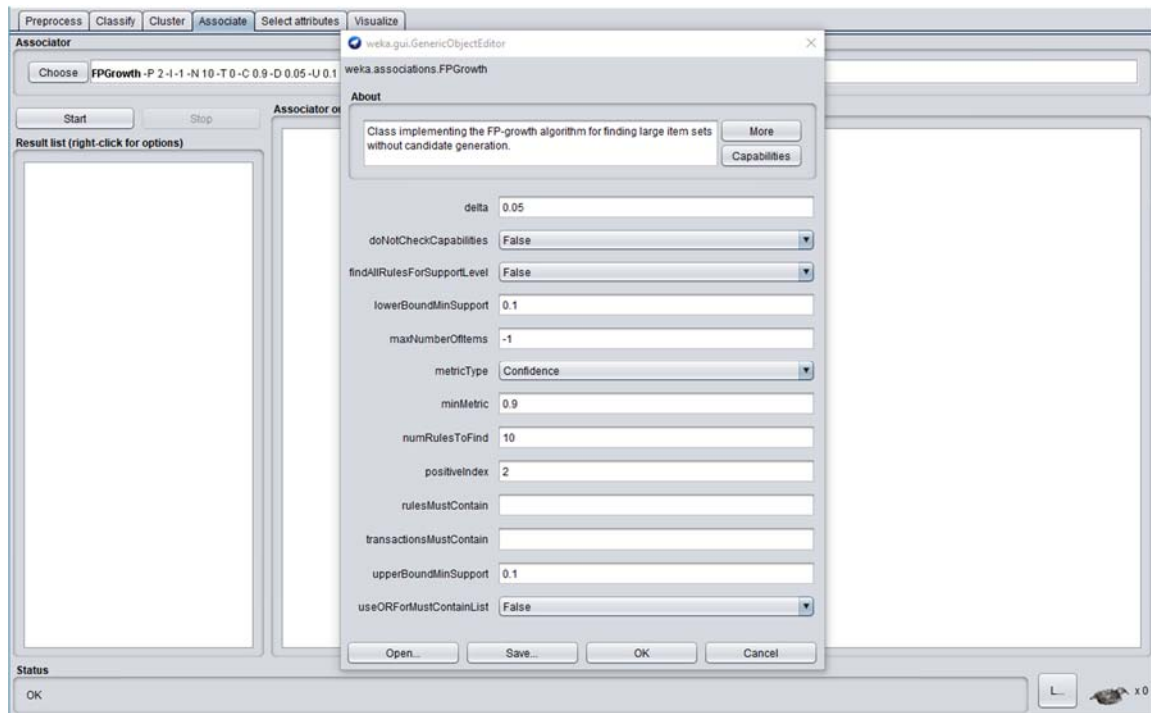
- Click on OK and run the algorithm again by clicking on the Start button. Do you observe a difference regarding the runtime in contrast to the Apriori algorithm? Is it faster or slower? How many rules does it find in total?

- Now, again go the parameters list by clicking next to FPGrowth. Change the values for the parameters upperBoundMinSupport and lowerBoundMinSupport to 1.0. Click on OK and run the algorithm again by clicking on the Start button. What do you observe? How many rules are found in total?

- Now play around with different parameter settings for metricType and the value for minMetric. You can also change the upperBoundMinSupport and lowerBoundMinSupport parameters if you want. Run at least three different settings of metricType and minMetric combinations. Do you observe any difference regarding runtime and the rules that are found?

Task 3: Classification

In this task, you will perform a classification task. To this end, you will load the dataset (1.1),
run a classification algorithm (1.2) and explore different algorithms and parameters (1.3).

3.1 Import the dataset

- Click on Open file … in the Preprocess tab and load the dataset *diabetes.arff* that we
  provided with this assignment. Make sure to select Arff as file format.

- You should now see the dataset with its two attributes and statistics about the dataset.
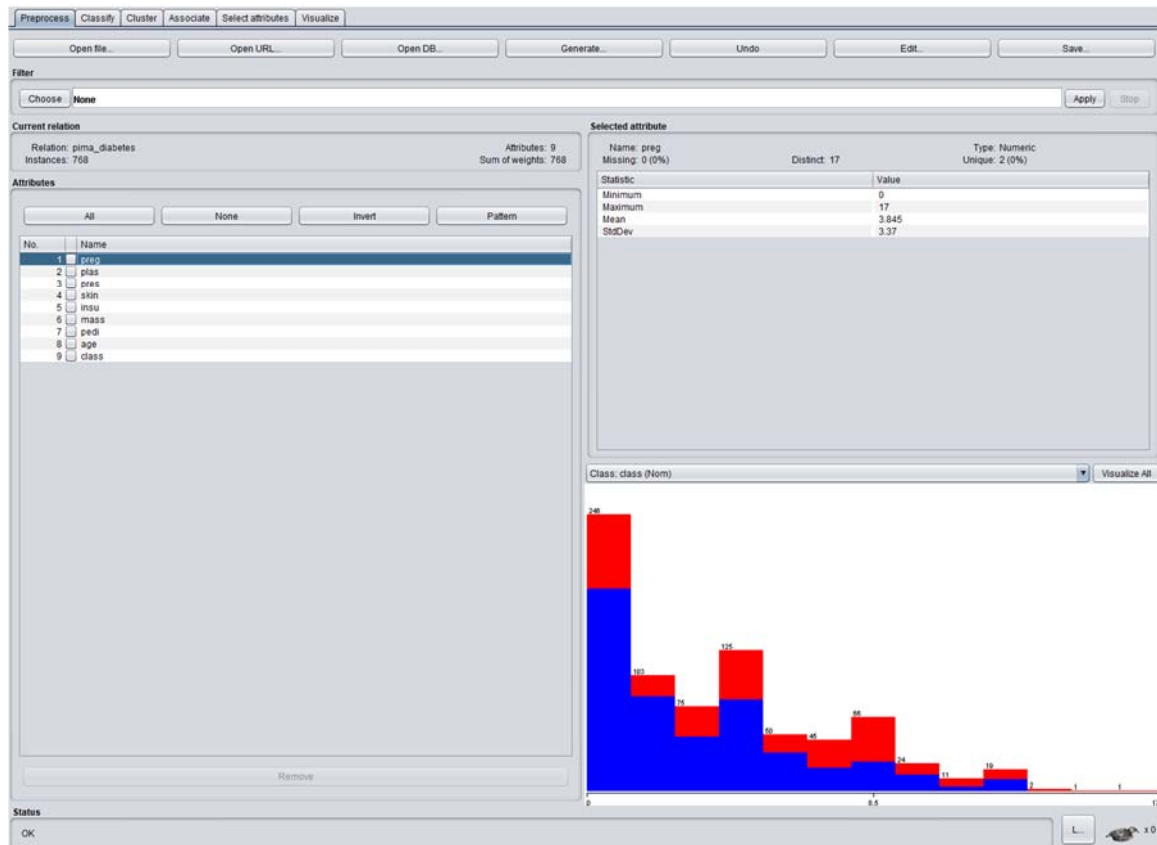
- In the Attributes section click on the attribute class to see more information about this
  attribute.

  *S2.1: Provide a screenshot of the result when you clicked on the <u>class</u> attribute
 (similar to the given screenshot but for the class attribute).*

  *Q3.1: What are the possible values for the class attribute and how often does each
  value occur? What do you think does this dataset describe (from just looking at
  the attributes, in particular on the class attribute, and the name of the dataset)?*

3.2 Run a Decision Tree on the imported dataset

- Click on the tab Classify. Click on Choose and then select J48 underneath the trees folder.
  This is a Java implementation of the C4.5 decision tree algorithm, which follows the basic
  generation steps of a decision like covered in the lecture. The splitting criterion is based
 on an entropy measure.
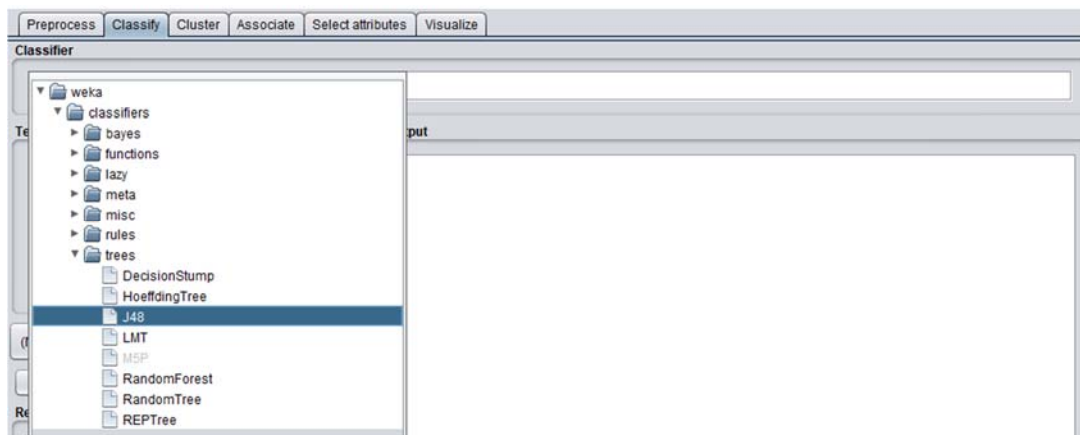
■ Click Start to run the algorithm. The output should look like the following:
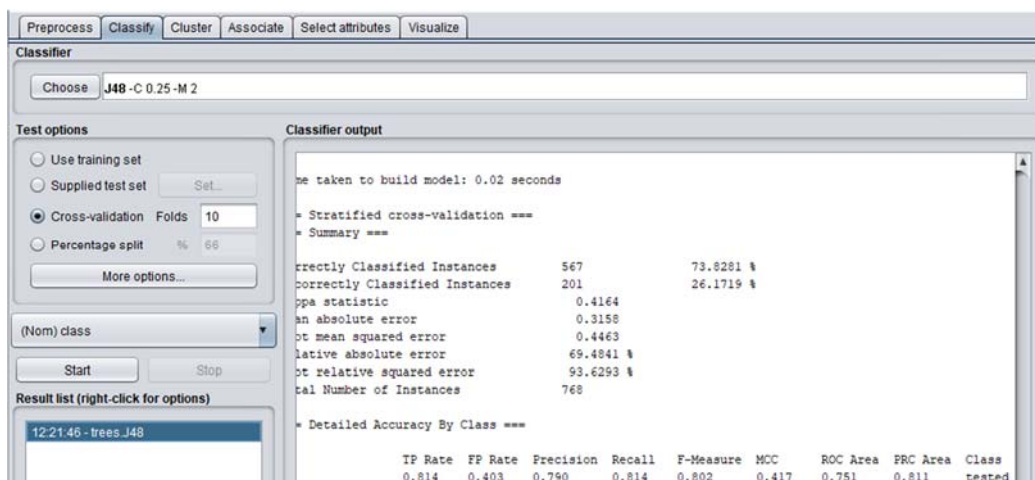
S2.2: Provide a screenshot of the whole console output that shows the Precision, Recall and F-Measure values for the weighted average.

Q3.2: What are the Precision, Recall and F-Measure values of your model? What do these values express? Report the results only for the weighted average.

■ In the Result list right-click on trees J48 and then choose Visualize tree.
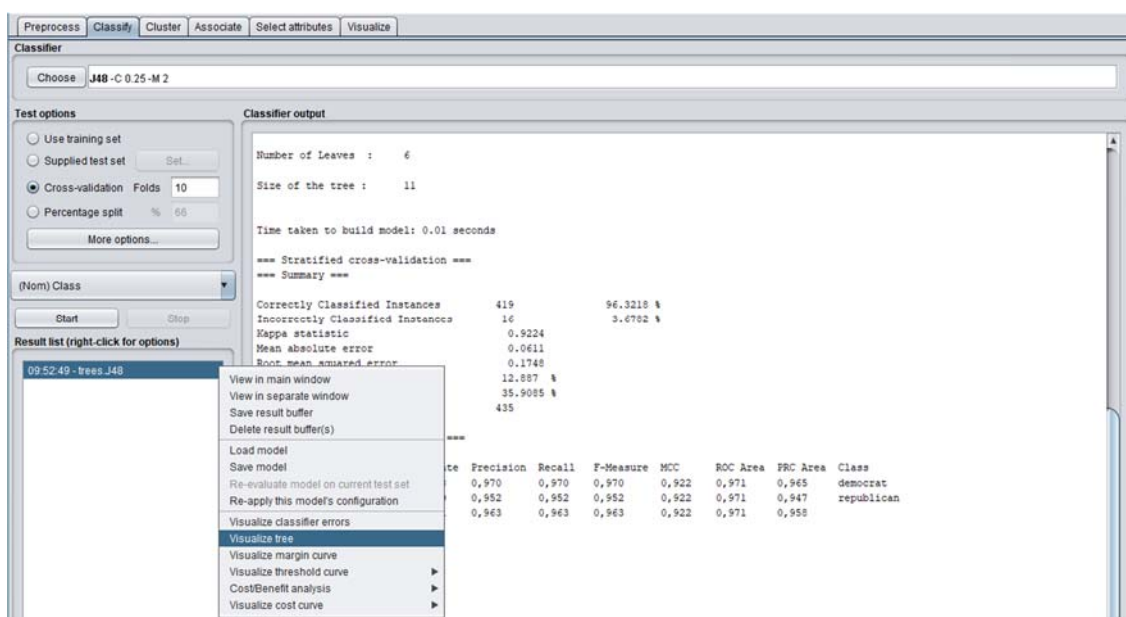
■ The resulting tree should look similar to the following tree. However, note that we only show the first three levels of the tree.
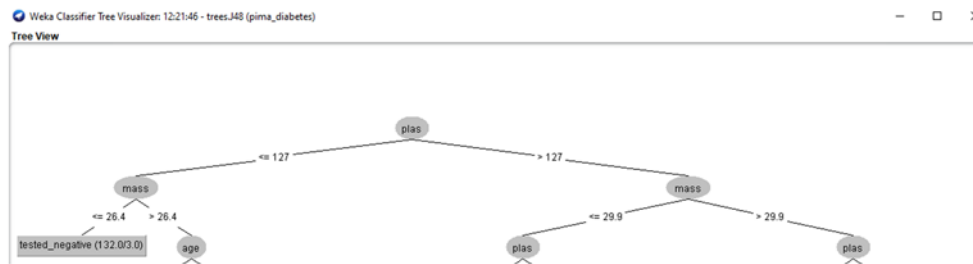
Q3.3: Where do you find the <u>classes in your tree</u>? What is the <u>depth of your tree</u>?

Q3.4: Find at least one rule for each of the classes in the dataset, i.e., a rule in which case a (new) sample is assigned to a class.

■ Assume you want to classify a new sample x which has the following values for the attributes:
plas=100, mass=33.1, pedi=0.57, preg=6, age=35, press=82, skin=50, insu=423, class=tested_negative

Q3.5: What class would x have according to your tree? Is this the correct class?

3.3 Explore different algorithms and parameters

- Click on the tab Classify. Click on Choose and then select RandomTree underneath the trees folder. Click on Start to run the algorithm.
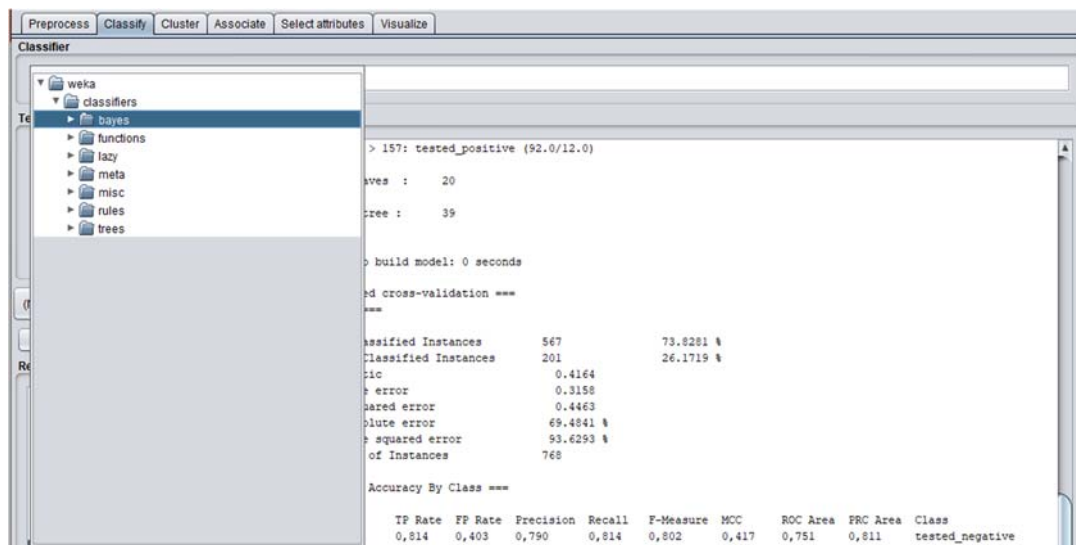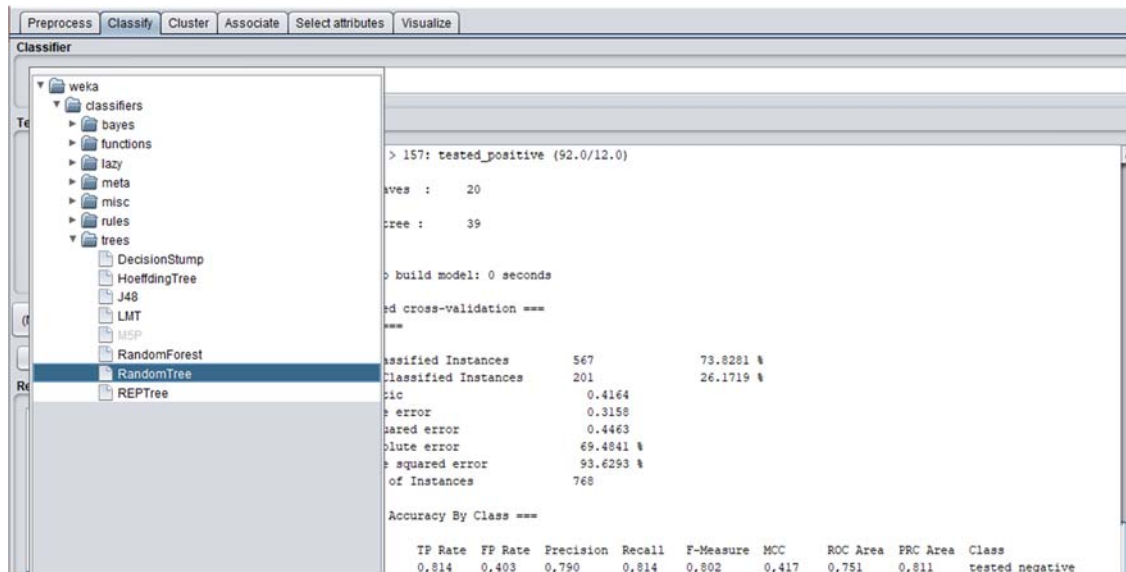

- What are the Precision, Recall and F-Measure values of the model? Are the results better or worse than with the J48 Tree?

- Now, visualize the tree. To this end, right-click on trees.RandomTree in the Result list and choose Visualize tree.

- How does the tree differ from the tree from Task 3.2? Is it larger or smaller? What do you think are the reasons for this?

- Next to Choose, click next to Random Tree to open the parameter list.
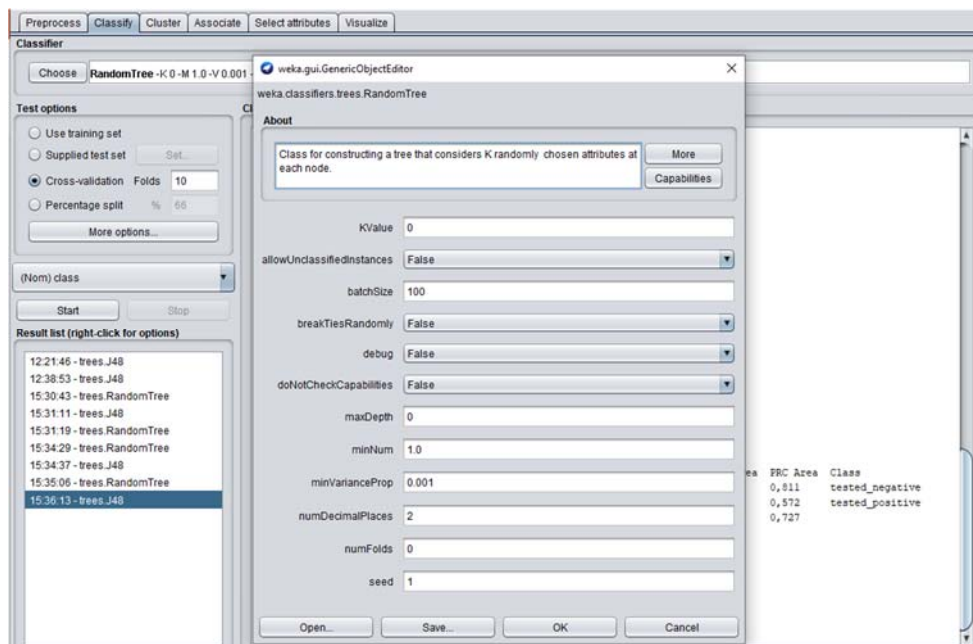
- What do you think, which parameters can you change to make the tree smaller/larger?

- Try it out and run the Random Tree with at least three different parameter settings.

- Now, try to increase the accuracy of your classification model. To this end, you should run different classification algorithms and test whether they provide more accuracy. Use

Precision, Recall and the F-Measure as accuracy measures. You should run <u>at least four</u> <u>different algorithms</u>, while at <u>least two</u> should be <u>other</u> kinds of classification algorithms <u>than</u> decision trees, i.e., you can choose from bayes, lazy, functions, meta, misc and rules. Document the results for each algorithm even if the accuracy does not increase.