Thema 9: Introduction to Data Mining

Introduction

In this quarter project you will get to know several aspects of Data Mining (DM), also called Machine Learning (ML). These two terms are not quite the same but are often used as if they are. By the end of the course, you will know what the difference is.

You will work individually on a ML problem of your own choice, but one or more other students will analyze the same dataset. This will make troubleshooting easier and opens the possibility of peer review (analysis and code), which is also an important skill to master: providing help and critical feedback to colleagues.

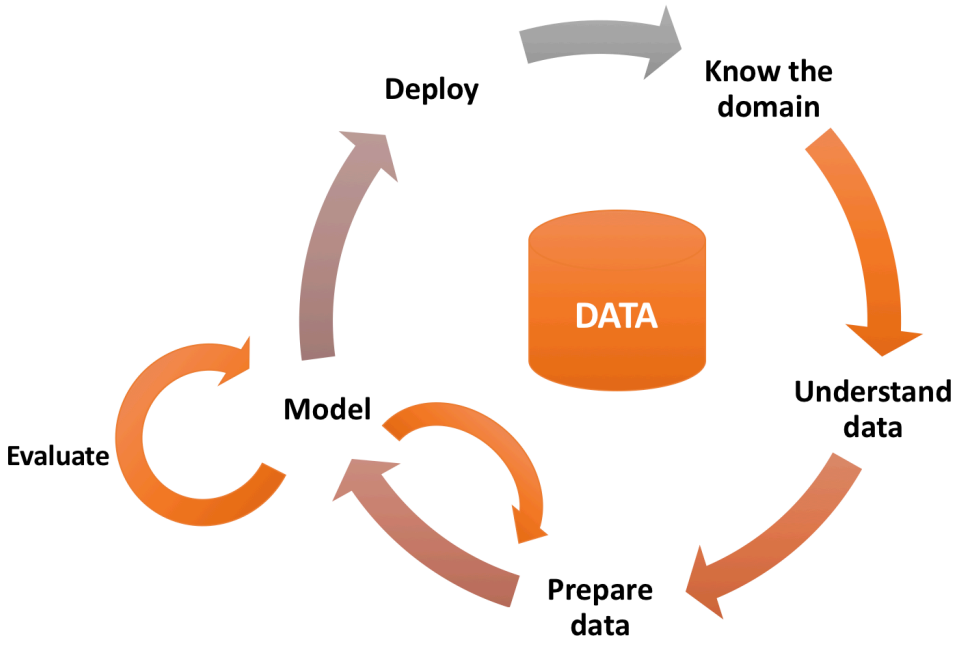
Contents, Learning outcomes & Assessment criteria

See Course Description("Vakomschrijving") here.

Course program

This is the first time this course is run. Please keep in mind the actual program may vary! For exact deadlines, submission details and peer review rules, refer to the Blackboard course. Make sure you have R, RStudio and Weka installed (also on your home computer).

In this project, we will explore the entire data mining life cycle:



****All** activities up to week 5 should be logged using an RMarkdown document with reproducibility and controllability as focus points.** This document will be reviewed by your teacher and (partly) by your fellow students (peer review) and should be maintained, together with all your code, in a Bitbucket repository.

****All** source code files should have a license note.** In Java, Python and plain R source files as a header at the top and with RMarkdown as a chunk below the or a. It should look like this:

```
/*
 * Copyright (c) 2018 <Your Name>.
 * Licensed under GPLv3. See gpl.md
 */
```

In other languages you use other comment characters of course. Always put a copy of the license in the root of your code folder. You can find it here: http://www.gnu.org/licenses/gpl.md. IntelliJ and PyCharm have tutorials on how to do this automatically: https://www.jetbrains.com/help/idea/copyright.html. We will discuss the implications of this in class.

Activities marked in bold and with the "package" icon represent deliverables that will be assessed by the teacher or fellow students. It looks like this:

Deliverable: at the end of the course, you will need to submit a log of your complete analysis in the form of a well-maintained RMarkdown document, as well as the pdf generated from it. When writing the log, always keep reproducibility of your research in mind.

Exact deadlines of the weekly assignments will be published on Blackboard or on the course website.

Week 1

Choose research topic and dataset

Choose a topic you would like to work on from this listing:

- UCI website (https://archive.ics.uci.edu/ml/index.php)
    - Breast cancer diagnosis: benign or malignant?: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29
    - Cervical cancer risk factors: https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29
    - Epileptic seizure recognition: https://archive.ics.uci.edu/ml/datasets/Epileptic+Seizure+Recognition
    - Yeast protein localization: https://archive.ics.uci.edu/ml/datasets/Yeast

    - Thyroid disease prediction: https://archive.ics.uci.edu/ml/datasets/Thyroid+Disease
    - Smartphone-Based Recognition of Human Activities and Postural Transitions: https://archive.ics.uci.edu/ml/datasets/Smartphone-Based+Recognition+of+Human+Activities+and+Postural+Transitions
    - Predicting hospital readmission of diabetes patients: https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008
- Kaggle https://www.kaggle.com/competitions
    - Predict bird group from bone measurements: https://www.kaggle.com/zhangjuefei/birds-bones-and-living-habits
- Or maybe a Hanze project is available…

It is possible to investigate some other dataset of your own choosing. Here are some criteria for it:

- It is a supervised learning problem (there are class labels for a training set)
- It already consists of textual/tabular data.
- It is a dataset from the life sciences in broader sense: biology, chemistry, medicine, and "quantified self" datasets are eligible.
- It has at least 5 attributes (but preferably more) and at least several hundred examples (but preferably more).
- With public datasets: there is at least one publication on the data or the corresponding research.

Your chosen project will need to be approved by the teacher.

Document status of knowledge and nature of the data

When you have selected a project to work on, find out where the data was originally published. Read this paper and summarize its findings. This will be the basis of your introduction of the report.

Describe carefully: how many attributes are there, what is the class attribute and what are its possible values, what data type have the attributes and what is their range (possible values) and distribution. In what way were the attributes measured/collected?

Formulate own research question and project goals

Given what you now know about the research you have chosen, formulate a research question you think is interesting and relevant. Here are some guidelines to help you formulate your own:

Questions should in some way. . .

Be worth investigating
Contribute knowledge & value to the field
Be valuable to society

Characteristics of a good research question:

The question is feasible (timeframe, technical and financial constraints).
The question is clear and precise (not too broad or narrow).
The question is measurable (answerable) – can it be supported or contradicted?

Also see this reference for writing research questions
https://cirt.gcu.edu/research/developmentresources/tutorials/question.

For instance, if you were to investigate the Wine Reviews dataset (see https://www.kaggle.com/zynicide/wine-reviews), a question could be "The goal of this project is to develop a machine learning model that can use the word composition of the wine description field to predict wine quality".

Your research question will be peer reviewed by other students
Submit your own research question in Blackboard and provide feedback to those of two others.
When reviewing other's research questions, consider the abovementioned characteristics. Use 150-200 words to give your feedback. Try to give positive and constructive feedback.

Week 2

Carry out an exploratory data analysis

Using R (and Weka), perform a so-called Exploratory Data Analysis (EDA). Use well-annotated tables and figures to present your results in your work RMarkdown log, and discuss your findings critically.

Consider these aspects:

- Are the classes evenly or unevenly represented?
- Are there missing data? If so, are there many, what is their relevance/impact and how do you propose to do deal with these instances and these missing values?
- Are there outliers? If so, what do you propose to do with them?
- Are there correlated -i.e. dependent- attributes? Describe these correlations, also to the class attribute. This is an especially important aspect, since many machine learning algorithms assume that all attributes are independent.
- Are there attributes that should be transformed? Think of log transform, numeric to nominal etc. Describe these.
- Create relevant plots and perform clustering (hierarchical, kMeans) and/or PCA and visualize these to discover visually apparent patterns that can help you understand your data and the classification process that will follow. Describe what you see.

Describe your general view of the quality of the dataset - is it good, does it seem corrupted, are there sufficient examples recorded, should other measurements have been taken? Are attributes independent?

Week 3

Create a clean dataset

Given your findings of last week, modify your dataset so that it is ready for machine learning experiments. This may involve transformations, removing instances, recoding attributes. This should of course all be logged.

Determine quality metrics relevant for your research

Read this Wikipedia page: https://en.wikipedia.org/wiki/Sensitivity_and_specificity (and/or the corresponding remarks in the book: page 180-181). When evaluating ML algorithm performance, accuracy is the default quality metric. However, for your particular application, other metrics may be more accurate or relevant. Reflect on these and describe which are most important for your project, why this is so, and how you can measure them.

Related but different are the performance aspects of speed, scalability, possibility of parallelization and feasibility of online classifications (as opposed to batch processing). Describe and define criteria that are important for your research. Also describe how you are going to determine and evaluate these aspects.

Your quality metrics will be peer reviewed by other students.
Submit your own research quality metrics on Blackboard and provide feedback to those of others, considering the above guidelines.
When reviewing quality metrics, consider these aspects:

1. **Do you agree with the chosen metrics, the reasoning behind the decisions, and the way there are going to be determined?**
2. **Can you think of other (better) ones?**

Classroom discussion on findings so far

Participate in the discussion and share your insights.

Week 4

Investigate performance of ML algorithms

Using your clean dataset, start investigating the performance of all standard ML algorithms. You should always include ZeroR and OneR to measure baseline performance. Besides these, you should include at the least include representatives of all classifier categories: NaiveBayes, SimpleLogistic, SMO, IBk (Nearest Neighbor), J48 (C4.5) and RandomForest. Carry out classifications using 10-fold cross validation, and record relevant quality metrics: speed, accuracy, TP, FP, TN, FN (the confusion matrix) and of course the quality metrics you have chosen yourself. It is quite easy to do this using the Weka Experimenter.
It probably most efficient to store these data in a table that you can read into R.

Present and discuss your findings and argue which line of research -i.e. which algorithm(s)- will probably be most effective to investigate further.

Use Weka Experimenter to optimize a selection of algorithms

Using the Weka experimenter, investigate the effect of different algorithm settings with the goal of improving algorithm performance. Also investigate the effect of Attribute Selection methods.
Apply appropriate statistical tests (the Experimenter supports this). Again, take into account the quality metrics you specified. Record, present and discuss your findings.

ROC and learning curve analysis

Create a ROC curve visualization of one or two final algorithms with optimal settings. Is the result satisfying? Take into account the quality metrics you have defined in an earlier stage. Create a learning curve as well. How much data do you need to get a reasonable performance estimate?

Week 5

Write Results and Conclusions

Settle on a final choice for algorithm and its optimal settings. Perform some last tweeks if you think this is required. Save the model as a serialized Weka/Java object.
Write the Results and Discussion & Conclusion sections of a scientific paper to publish your results. Results should include data exploration and cleaning, and of course the search for an optimal classifier for your project. Liven up your results section using (well annotated!) figures. Write a discussion section and conclude whether you are confident you found the best possible classifier. Describe possible future work to improve the classifier or find a better one. Of course, if your research log was well maintained, writing these chapters won't be that much of a challenge!

Using Blackboard, submit the Results and Discussion & Conclusion paper for review by the teacher. It is supposed to be 3 – 5 pages long.

Create Java wrapper program for your learned model

OK, so you have a nice model - so what. You have to publish it in a user-friendly way so that others can use your model to predict the class of new, unknown instances. That is why you now have to create a command-line Java application that wraps your final optimized model. This will make it possible to quickly classify new instances. The program should be able to classify new instances having the required attributes – not necessarily all attributes that were present in the original dataset! The program should be able to classify single instances fed from the command line, or batches fed through an input file. Maybe your model should even be able to perform online (streaming) classification? As a bonus, your program could be extended to give some classification statistics when working batch-wise, or to output probabilities instead (or as well as) class labels.

Use the Weka API. This repo contains some demo code on how to get going with building your program around a serialized Weka classifier:

https://bitbucket.org/minoba/wekaapidemo/overview. Use the Apache CLI API for processing command-line arguments (see http://commons.apache.org/proper/commons-cli/ and a demo program at https://bitbucket.org/minoba/clidemo. This is also an example on how to start and build a Gradle-managed Java project in IntelliJ.

Week 6

Continue Java wrapper program

Week 7

Finish wrapper program

Finish your command-line application and any loose ends that may be present. Make sure your program is on Bitbucket and has a good readme.md describing your algorithm, what it can be used for, and how to install (set up) and run your program. Do NOT forget to describe all possible command-line arguments and their possible values (and what they default to), and of course the dataformat your program should be fed with. I encourage you to do a Google search and have a look at some good Readme.md examples.

Pitch your results

Suppose you are working for a big firm. You have 90 seconds with the CEO. You have to pitch your work and convince him your work is really worthwhile to continue, otherwise you will lose your job.
You will have to do this in the classroom, without presentation materials (powerpoint) or other digital support.

Finalize report

The teacher has provided you with feedback on the Results and Discussion sections of your scientific paper. Use this feedback to improve it and submit a final version. Also add a description of a possible minor project to be carried out in one of the bioinformatics minors (Application Design or High performance / High throughput Biocomputing. This should include a goal, purpose and a deliverable. Can you maybe even think of (commercial) partners that may be interested in this classifier?

Submit your final paper, your research log -both as Markdown and as Pdf- and a link to your -public- Bitbucket repo (that also has an executable in the download section).

The final paper should have these contents (pages are an approximation):
Introduction (½ page) with your research goal (improved with feedback from peer review)
Materials & Methods (1 page)
Results (5 pages maximum)
Discussion & Conclusions (1 page)
Project proposal for minor (½ page)
References