# Take-home exam
# DIT866/DAT340: Applied Machine Learning, March 17–18, 2019

**Course responsible:** Richard Johansson, CSE (richard.johansson@gu.se, +46317721887)

**Please note:**

- To keep grading anonymous, please do not include your name in the file you submit.
- If there is something you don't understand about a question, please contact Richard over email or phone (9 AM – 5 PM) as soon as possible. No answers guaranteed after 5 PM on Monday.
- If you find typos or errors, please let me know and I will post an updated version as soon as I can.
- Until the submission deadline, it is strictly prohibited to communicate with other students about the contents of the take-home exam.
- Standard plagiarism regulations apply and the submitted file will be checked by a plagiarism detection program. Your solutions need to be your own and you are not allowed to copy any material from any source. (Please get in touch if you are unsure.)

# Part 1: Basic questions

You need a score of at least 27 points in this part to receive a passing grade (*G/3*).

## Question 1 of 12: Master's program admission (9 points)

At a university, planning the staffing of future courses may be quite challenging because of the difficulty of predicting how many of the students who are admitted to the university's programs will actually be present when the first semester starts. When selecting students who are applying to a Master's program, we would like to apply a predictive model that tries to determine whether a given student is likely to turn up or not.

**(a, 6p)** Explain how you would implement a machine learning model that would solve this prediction task. You don't need to show Python code, but please give a description of the system and explain all steps you would carry out when developing it. In this description, mention at least four *features* that you think could be useful in your system; you may assume that a suitable amount of historical data is available, and that the features you suggested can be extracted automatically from this data. But please only include features that you think may be available in a realistic scenario.

**(b, 3p)** How would you propose to evaluate a system of this type? Let's say we apply your system, evaluate and get the following result:

- student ***does*** show up, the system thinks the student ***will*** show up: 13,278 students
- student ***does*** show up, the system thinks the student ***will not*** show up: 1,655 students
- student ***does not*** show up, the system thinks the student ***will*** show up: 2,298 students
- student ***does not*** show up, the system thinks the student ***will not*** show up: 4,874 students

Compute the evaluation score based on the result above and your proposed evaluation procedure.

## Question 2 of 12: Linear and neural network regression (6 points)

In machine learning toolkits, there are several types of models that predict numerical outputs (regression models), including *linear* models and *neural network* models. For instance, scikit-learn has a few different types of linear regression models (`LinearRegression` and others), and one type of neural network regression model (`MLPRegressor`).

**(a, 2p)** When is it preferable to use neural network regression instead of linear regression?

**(b, 2p)** What advantages do linear models have over neural network models?

**(c, 2p)** In what way can we say that neural network regression *generalizes* linear regression: or conversely, that linear regression is a special case of neural networks?

## Question 3 of 12: Spotting objects at sea (6 points)

A shipping company operating in the north Atlantic region wants to develop an automatic system that uses a camera to detect and classify large objects at sea. The company has installed a camera in each of its ships that is going to be used for this purpose. They have also set up a web-based system where a person can mark the relevant part of an image, as in the following example:

They have defined a set of predefined categories of objects, including *iceberg* such as in the example above, as well as *ship*, *whale*, *oil rig* and some other categories.

**(a, 3p)** Describe how you would recommend the company to collect the images for the training data for this classifier.

**(b, 3p)** For creating the training data, let's assume that you have some people who look at images and manually tag various types of objects in them. How would you investigate whether this training data production process is reliable?

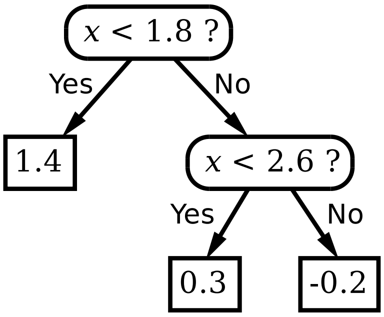
## Question 4 of 12: Model degradation (4 points)

It is quite common after deploying a trained model that it "degrades" over time: its performance becomes worse and worse as time goes by. What is the reason for this?

## Question 5 of 12: Decision tree models for regression (7 points)

Please answer the following questions about decision tree models and tree ensembles.

**(a, 2p)** We have trained a decision tree model that predicts the numerical output $y$ as a function of a single input variable $x$. Here is the tree:



Please draw a plot of the decision tree's predictions for different values of $x$.

**(b, 2p)** Generally speaking, how will the shape of the curve look if we allow the tree to be deeper, for instance by increasing the value of `max_depth` or `max_leaf_nodes` when training a `DecisionTreeRegressor`? (This of course depends on what training data we have, but what can we say in general?)

**(c, 3p)** When training a random forest, such as a `RandomForestRegressor` in scikit-learn, the base models are decision trees. The number of decision trees is controlled by the hyperparameter `n_estimators`. If we train a small random forest consisting of two decision trees, how do these decision trees differ in terms of the *training set* they have been exposed to? What is the purpose of this?

## Question 6 of 12: Age prediction (8 points)

Your company wants to develop a system that automatically determines the age of a person in a photograph. The company collects a large dataset of pictures of people for which the age is known.

**(a, 3p)** What type of machine learning model would you use and how would you train it? You may assume that there is exactly one person in each photograph.

**(b, 1p)** How would you evaluate the predictions?

**(c, 4p)** The Swedish Migration Agency (*Migrationsverket*) buys the system from your company, for the purpose of automatically determining the age of young asylum seekers. Discuss what may go wrong.

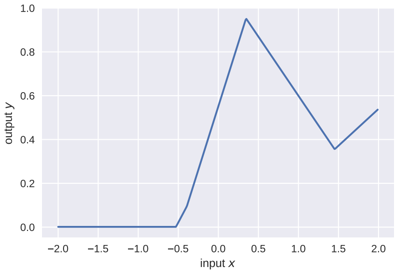
# Part 2: Questions for the high grades

DIT866: You need a total score of **63** points to receive the grade *VG*.
DAT340: You need a total score of **50** for the grade *4*, and **63** for the grade *5*.

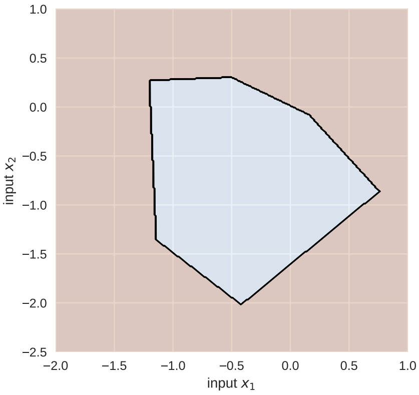
## Question 7 of 12: Neural networks (5 points)

It is possible to use neural networks for classification and regression; for instance, in scikit-learn we have `MLPClassifier` and `MLPRegressor`. It is quite common to use *rectified linear units* (ReLUs) as the activation function in the hidden layer.

Let's say that we build a neural network regression model for a one-dimensional input and that we use ReLUs in its hidden layer. We then plot the model's predictions as a function of the input. In this case, the plot will consist of separate sections where the plot is a completely straight line, and there will be a sharp corner in the plot when we come to a new section. For instance, here is an example:



Similarly, for a classification model, the decision boundary will have a shape consisting of straight line segments and sharp corners if the hidden layer uses ReLUs. For instance, if we train a binary ReLU-based neural network classifier using two-dimensional inputs, we might get a decision boundary like this:



If we use other activation functions, such as the hyperbolic tangent, there are no straight line segments or sharp corners. Why do we see this shape when we are using ReLUs in the hidden layer? For simplicity of explanation, you may assume that there is just one hidden layer.

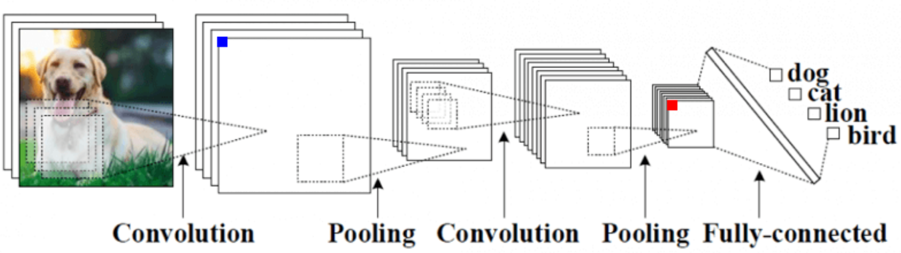
## Question 8 of 12: News recommendation and political polarization (5 points)

It has been argued that the introduction of news recommendation systems can cause a clustering of users, and that this leads to an increased level of political polarization in society. For instance, according to this view, people who tend to view online material with a left-wing orientation will do so to an even larger extent after a recommender system has been introduced. There have also been claims that news recommenders will tend to expose users to articles with a more extreme angle than what they would otherwise prefer; for instance, that people who have viewed some articles with a center-right orientation will be recommended far-right material after the recommender system has been running for some time.

Please try to explain why recommender systems for news may have these effects.

## Question 9 of 12: Convolutional neural networks (5 points)

Consider a convolutional neural network for images that has the following structure:



If we are using Keras, the code to declare the structure of the CNN might be as follows:

```
model = Sequential()
model.add(Conv2D(32, kernel_size=(5, 5), activation='relu',
                 input_shape=(img_width, img_height, 1)))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Conv2D(64, (5, 5), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(number_of_classes, activation='softmax'))
```

**(a, 2p)** After applying the first convolutional layer, the result is a number of feature maps. What does the first element in the first feature map represent? (This corresponds to the blue square in the figure.) How is it computed?

**(b, 3p)** We then apply a pooling layer, another convolutional layer, and another pooling layer. Consider the first element in the first feature map after these steps. (This element is represented by the red square in the figure.) What parts of the original image have been involved in the computations that resulted in this element?

## Question 10 of 12: Training a linear regression model (8 points)

We'd like to train a linear regression model based on the following loss function, which is known as the "epsilon tube":

$$\text{Loss}(\boldsymbol{w}, \boldsymbol{x}, y) = \max(0, |y - \boldsymbol{w} \cdot \boldsymbol{x}| - \varepsilon)$$

As usual, in this formula $\boldsymbol{x}$ is a feature vector representing the instance for which we are making a prediction, $y$ is the desired output for this instance, and $\boldsymbol{w}$ is the model's weight vector. $\varepsilon$ is a hyperparameter set by the user that is greater than 0.

The subgradient of this loss function with respect to the weight vector $\boldsymbol{w}$ is

$$\nabla_{\boldsymbol{w}}\text{Loss} = \begin{cases} \boldsymbol{x} & \text{if } \boldsymbol{w} \cdot \boldsymbol{x} - y > \varepsilon \\ -\boldsymbol{x} & \text{if } y - \boldsymbol{w} \cdot \boldsymbol{x} > \varepsilon \\ (0, \ldots, 0) & \text{otherwise} \end{cases}$$

**(a, 1p)** Describe the shape of this loss function in words and/or graphically. Where is it zero, where does it increase?

**(b, 4p)** Write the pseudocode (or Python approximation) for an algorithm to train a regression model by minimizing this loss function on a training set.

**(c, 3p)** Add an $L_2$ regularizer to the model. How does this change your pseudocode?

## Question 11 of 12: Classifiers with restricted outputs (5 points)

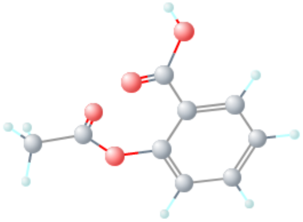
In supervised classification, our machine learning model selects the output $y$ for a given input $\boldsymbol{x}$ from a finite set $\mathcal{Y}$ of discrete categories. In general, for a given input the classifier is allowed to pick any item from the output set $\mathcal{Y}$, but in some applications the classifier may be *restricted*: it is only allowed to select from a subset of $\mathcal{Y}$. This restriction may depend on properties of the input $\boldsymbol{x}$ or on the circumstances where the classifier is applied. In some cases, this type of restriction may make the classifier more reliable, or it may make classification more efficient if $\mathcal{Y}$ is very large but not every possible output needs to be considered.

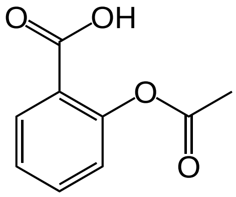
To give you a contrived example, assume that we are building a classifier of images of pets, and $\mathcal{Y} = \{cat, dog, rabbit, \ldots, reindeer, goldfish, iguana\}$. For one instance $\boldsymbol{x}$, we may have some advance knowledge that we are dealing with a small furry animal, so the allowed subset is $\{cat, dog, rabbit\}$.

Let's assume that there is a function called $\textsc{LegalOutputs}(\boldsymbol{x})$ that returns the subset of $\mathcal{Y}$ that is allowed for an instance $\boldsymbol{x}$. How would you build a classifier that never returns an output that is not allowed?

## Question 12 of 12: Machine learning and molecules (8 points)

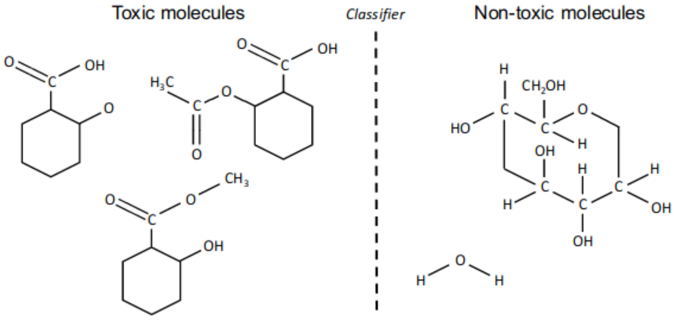
In chemistry and biology, the structure of a *molecule* can be expressed and visualized in different ways. For instance, the aspirin (or acetylsalicylic acid) molecule can be drawn as a "ball-and-stick model" (left) or as an equivalent skeletal formula (right). In the left figure, the four red balls are oxygen, the nine grey balls are carbon, and the eight small blue-green balls hydrogen. For brevity, the skeletal formula does not explicitly write out carbon–hydrogen combinations. Double connections in both figures correspond to covalent bonds.





There are also many ways to express molecules as strings. One of the simplest string representations is the molecular formula that is probably familiar to you from chemistry classes. This string representation just lists the numbers of each type of atom, such as $C_9H_8O_4$ for the aspirin molecule. The molecular formula says nothing about the structure of the molecule, but there are more complex string representations that encode the structural information. For instance, SMILES is one such string representation. The SMILES string corresponding to the aspirin molecule is `CC(=O)OC1=CC=CC=C1C(=O)O`.

**(a, 6p)** We would like to develop supervised classifiers that distinguish between different categories of molecules. For instance, we may want to classify molecules as toxic or not:



Other properties that we may want to use for classification include the solubility or thermal conductivity.

How would you develop a model that classifies molecules? Propose *two* solutions and argue for which of them you prefer. You may assume that you have a labeled training set that is "large enough" and that each molecule in the dataset is expressed in a representation that is expressive enough for your purposes.

**(b, 2p)** Give at least one example of another classification or regression problem *not* related to chemistry or biology where the input to the predictive model is a graph. (If you have no idea, try to be inventive.)