Written examination
DIT865/DAT340: Applied Machine Learning, March 15, 2018

Course responsible: Richard Johansson, CSE (+46 31 772 1887)

Allowed accessories: Calculator, written notes (**one** A4 paper)

Please note:

- If there is something you don't understand about a question, please ask the course responsible to clarify when he comes to the exam room (at about 9:30 and 11:30).
- Make sure that your handwriting is legible. You will get no points for unreadable solutions.
- If your solution to a question is incomplete, please turn it in anyway! Every point counts.

Part 1: Basic questions

You need a score of 32 points in this part to receive a passing grade (*G/3*).

Question 1 of 12: Predicting house prices (8 points)

A real estate firm would like to build a system that predicts the sale prices of a house. They create a spreadsheet containing information about 1,460 house sales in the Gothenburg area. In addition to the price, there are 79 features describing the house, such as *number of bedrooms*, *total indoor area*, *lot area*, *has a garage*, *location*, etc.

(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.

(b, 2p) Explain why the model you built is probably useless in the long run.

Question 2 of 12: Predicting customer churn (8 points)

A mobile phone service provider would like to build a model that predicts whether a subscriber is likely to cancel the subscription in the next couple of months. (The jargon term for this is customer *churn*.) The purpose of this classifier is that the company wants to send these subscribers some special offers in order to dissuade them from dropping their subscriptions.

The company makes a training set by collecting some historical data, and then they implement a classifier using scikit-learn. This is how they declare their classifier:

```
pipeline = make_pipeline(
    DictVectorizer(),
    StandardScaler(),
    SelectKBest(k=100),
    LogisticRegression()
)
```

Can you describe the purpose of each of the four steps in the pipeline?

Question 3 of 12: Evaluation (8 points)

A telecom company has developed a classifier for detecting whether a cell-phone network base station is faulty or not.

(a, 3p) We evaluate the classifier on a test set. Here is the confusion matrix. (In the table, F means *faulty* and NF *not faulty*.)

		Predicted	
		F	NF
Truth	F	10	5
	NF	20	965

Compute the accuracy of the classifier.

(b, 2p) What is the accuracy of a majority-class baseline? (The class *not faulty* is the most common in the training set.)

(c, 3p) Would you say that the classifier is more useful than the majority-class baseline? Explain why or why not.

Question 4 of 12: Overfitting (6 points)

(a, 3p) What is *overfitting* and why is it a problem?

(b, 3p) Give an example of a method to reduce the risk of overfitting.

Question 5 of 12: Decision tree classifiers (4 points)

We have a training set that includes three features, and we'd like to predict a binary output variable. Here is the whole training set.

x_1	x_2	x_3	y
A	X	P	True
A	X	Q	True
A	Y	P	False
A	Y	Q	False
B	X	P	True
B	X	Q	False
B	Y	P	False
B	Y	Q	False

If we train a decision tree classifier using this training set, which feature would be considered at the first "branch" of the decision tree (the top node)? Please explain why.

Question 6 of 12: Pest control (6 points)

You are contacted by a food processing company that wants you to develop a classifier that detects whether a rat is present in an image. You collect a large dataset of images by crawling the web, and have annotators determine which images contain rats. This set of images can then be used as the training set for your classifier.

(a, 2p) Suggest a machine learning method to use for this classification task.

(b, 4p) After you have delivered your solution to the company, they get back to you and complain that when they evaluate on a new test set, they get precision and recall values that are much lower than what you reported to them. Explain what might have gone wrong.

Part 2: Questions for the high grades

DIT865: You need a score of 30 points in this part to receive the grade *VG*.
DAT340: You need a score of 16 for the grade *4*, and 30 for the grade *5*.

Question 7 of 12: Neural network regression (5 points)

We'd like to train a regression model that is able to predict a numerical output y, given a numerical input x. The inputs are one-dimensional (that is, each x is just one number). Using scikit-learn, we train a neural network:

```
MLPRegressor(activation='relu', hidden_layer_sizes=2)
```

After training, the weight matrices for the two layers are

$$\begin{bmatrix} 1 \\ -2 \end{bmatrix} \qquad \begin{bmatrix} -1 & 1 \end{bmatrix}$$

(To clarify: the left matrix contains the weights for the two hidden units, and the right matrix the weigths for the output unit.) Can you plot the output of the model as x ranges from -2 to 2?

Question 8 of 12: A variant of the perceptron (5 points)

The following pseudocode shows a variant of the perceptron learning algorithm. This algorithm often works significantly better than the standard perceptron algorithm. Can you think of a reason why this is the case?

$\boldsymbol{w} = (0, \ldots, 0)$
$\boldsymbol{w}_s = (0, \ldots, 0)$
repeat N times
  for $(\boldsymbol{x}_i, y_i)$ in the training set
    if $y_i \cdot \boldsymbol{w} \cdot \boldsymbol{x}_i \leq 0$
      $\boldsymbol{w} = \boldsymbol{w} + y_i \cdot \boldsymbol{x}_i$
      $\boldsymbol{w}_s = \boldsymbol{w}_s + \boldsymbol{w}$
return $\boldsymbol{w}_s / (N \cdot T)$

In the pseudocode, N is the number of epochs and T is the size of the training set.

Question 9 of 12: Logistic regression (8 points)

Logistic regression (LR) is one of the most popular machine learning models. Training a LR model is done by finding the weight vector $\boldsymbol{w}$ that minimizes the function f in the following equation:

$$f(\boldsymbol{w}, \boldsymbol{X}, Y) = \sum_{i=1}^{n} L(\boldsymbol{w}, \boldsymbol{x}_i, y_i) + \frac{\lambda}{2} \cdot \|\boldsymbol{w}\|^2$$

As usual, $\boldsymbol{X}$ is a list of feature vectors $\boldsymbol{x}_i$ of all the instances in the training set and Y the corresponding outputs y_i (each output coded as +1 or -1). λ is a user-defined parameter.

The loss function L is defined

$$L(\boldsymbol{w}, \boldsymbol{x}_i, y_i) = -\log \frac{1}{1 + \exp(-y_i \cdot (\boldsymbol{w} \cdot \boldsymbol{x}_i))}.$$

(a, 4p) Explain why the loss function looks like it does.

(b, 4p) What happens to $\boldsymbol{w}$ if we replace $\|\boldsymbol{w}\|^2$ by

$$\|\boldsymbol{w}\|_1 = |w_1| + \ldots + |w_n|$$

in the objective function f above?

Question 10 of 12: Word embeddings (8 points)

(a, 3p) What does it mean to *pre-train* word embeddings and why is this useful?

(b, 3p) It has been noted that word embeddings can encode some gender and ethnic biases.[1] Can you explain why this is the case?

(c, 2p) Can you think of an application where it may be useful to train "word embeddings," but the data is not text? (That is, the embedded objects are not words.)

Question 11 of 12: Sequence models (6 points)

(a, 3p) What does the notion of *state* mean for recurrent neural networks?

(b, 3p) What are the *encoder* and *decoder* in a neural machine translation system?

Question 12 of 12: Recommending fashion items (8 points)

An e-commerce platform that sells clothes wants to develop an image-based recommender system. Here are some examples of images of the company's items:

The image is the only piece of information that is available about each item. For customers, you have access to the purchase history but no other information.

On a high level, can you describe how you would implement such a recommender system?

[1] See for instance Bolukbasi et al. (2016), *Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings*, in Proceedings of Neural Information Processing Systems (NIPS).