ADVANCED MACHINE LEARNING

Question set

April 2021

Remark 1: Deadline for delivering the questions and the report of practical assignment: **23.59, 21 May 2021**. Upload the solutions in a pdf file to a task in poliformat.

Remark 2: The marks will depend on the depth of the solutions and comments. The presentation of the exercises will be also taken into account.

Remark 3: These exercises are individual exercises. The professor could ask for some clarifications in oral sessions upon request.

Theoretical exercises

Question 1 (1 point)

Given two distribution p and q, demonstrate if $D(p||q) + D(q||p)$ is or is not symmetric and satisfy the triangle inequality.

Question 2 (1 point)

For a given value of $Y = y$, is it possible that $H(X|Y = y) \geq H(X)$? Provide a demonstration.

Question 3 (2 points)

Complete example in page 22 in the slides to compute $H_{\mathcal{A}}(\theta|w)$. Define a new string with length 6 that includes the prefix "aba" and compute the table.

Question 4 (2 points)

Given the example in page 39 of the slides, carry out exercise 1 to compute the increasing of λ_1, or λ_2 or λ_3 (just two of them).

Question 5 (2 points)

Repeat experiment in page 61 of the slides but with a training sample of size 1000. Explain your results and the conclusions. Be concise.

Practical exercises: Learning of PCFG with the MMI algorithm

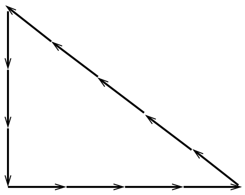
Toolkit description

This practical part assumes that you are familiar with Probabilistic Context-Free Grammars (PCFG). The toolkit includes a program for estimating a PCFG with the Inside-Outside algorithm, with the Viterbi-Score algorithm, with the MMI algorithm, with and without bracketed samples. As a simple application, a task on triangles has been defined.

PCFG can be used to represent geometric figures. Consider the following primitives:

$$a = \nearrow, \quad b = \rightarrow, \quad c = \searrow, \quad d = \downarrow, e = \swarrow, \quad f = \leftarrow, \quad g = \nwarrow, \quad h = \uparrow .$$

Then, a right triangle like this:



can be represented with the string *dddbbbbggggg*. Triangles with the sides sorted in a different way are possible.

In this task we will work with three types of triangles: right triangles, equilateral triangles, and isoscalen triangles. These type of triangles exhibit some relations that PCFG may capture. For this purpose, bracketed samples can be used to represent the relations. For example, the following property can be considered for right triangles: $G^2 = D^2 + B^2$, where G, D and B are respectively the number of g, d, and b in the string. The previous restriction is *represented* in the sample as: $(ddd)(bbbb))(ggggg)$.

The following commands perform a simple experiment for given training and test sets:

```
# train models
scfg-toolkit/scfg_learn_mmi -g MODELS/G-1 -f MODELS/right-0.10 -p DATA/Tr-right -n DATA/Tr-right-neg -H 0.1 -i 1
scfg-toolkit/scfg_learn_mmi -g MODELS/G-1 -f MODELS/equil-0.10 -p DATA/Tr-equil -n DATA/Tr-equil-neg -H 0.1 -i 1
scfg-toolkit/scfg_learn_mmi -g MODELS/G-1 -f MODELS/isosc-0.10 -p DATA/Tr-isosc -n DATA/Tr-isosc-neg -H 0.1 -i 1
# classify with the trained models and get results
scfg-toolkit/scfg_prob -g MODELS/right-0.10 -m DATA/Ts-right > r
scfg-toolkit/scfg_prob -g MODELS/equil-0.10 -m DATA/Ts-right > e
scfg-toolkit/scfg_prob -g MODELS/isosc-0.10 -m DATA/Ts-right > i
paste r e i | awk '{m=$1;argm="right"; if ($2>m) {m=$2;argm="equil";}
              if ($3>m) {m=$3;argm="isosc";}printf("right %s\n",argm);}' > results
scfg-toolkit/scfg_prob -g MODELS/right-0.10 -m DATA/Ts-equil > r
scfg-toolkit/scfg_prob -g MODELS/equil-0.10 -m DATA/Ts-equil > e
scfg-toolkit/scfg_prob -g MODELS/isosc-0.10 -m DATA/Ts-equil > i
paste r e i | awk '{m=$2;argm="equil"; if ($1>m) {m=$1;argm="right";}
              if ($3>m) {m=$3;argm="isosc";} printf("equil %s\n",argm);}' >> results
scfg-toolkit/scfg_prob -g MODELS/right-0.10 -m DATA/Ts-isosc > r
scfg-toolkit/scfg_prob -g MODELS/equil-0.10 -m DATA/Ts-isosc > e
scfg-toolkit/scfg_prob -g MODELS/isosc-0.10 -m DATA/Ts-isosc > i
paste r e i | awk '{m=$3;argm="isosc"; if ($1>m) {m=$1;argm="right";}
              if ($2>m) {m=$2;argm="equil";} printf("isosc %s\n",argm);}' >> results

cat results | scfg-toolkit/confus
#       equi isos righ  Err Err%
# equi  100    0    0    0  0.0
# isos   47   48    5   52 52.0
# righ    0    8   92    8  8.0
#
# Error: 60/300 = 20.00%
```

Adjusting the parameters, the error can decrease at least until 20%.

Question 6 (2 points)

Try to obtain the best possible results in the previous task (the mark in this exercise will depend a lot on the obtained results). The results could be checked by the professor if the experiments does not look fair. Explain your work and provide some conclusions. Hints: increase the number of training samples, both negative and/or positive samples.