KTH Computer Science
and Communication

Exam in DD2431 Machine Learning
2016-10-28, kl 14.00 – 18.00

Aids allowed: *calculator*, *language dictionary*.

In order to pass, you have to fulfill the requirements defined both in the A- and in the B-section.

A Questions on essential concepts

Note: As a prerequisite for passing you must give the correct answer to almost *all* questions. Only *one* error will be accepted, so pay good attention here.

A-1 Probabilistic Learning

In maximum likelihood estimation we adjust the model parameters so that:

a) The probability of the data given the model is maximized.

b) The training data and the test data have similar distributions.

c) The classes are linearly separable.

Solution: a

A-2 Regression and Classification

Choose the most proper statement reflecting the output formats of regression and classification.

a) They are both discrete.

b) Discrete for classification and continuous-valued for regression.

c) They are both continuous-valued.

Solution: b

A-3 Shannon Entropy

Consider a single toss of fair coin. Regarding the uncertainty of the outcome {head, tail}:

a) The entropy is zero.

b) The entropy is equal to one bit.

c) The entropy has nothing to do with the measure of uncertainty.

Solution: b

A-4 Overfitting

You have trained a model (classifier) using some training sample data. Under which conditions is overfitting most likely to occur?

a) Relatively complex model is used where few training samples are available.

b) Relatively many training samples are used.

c) Relatively simple model is used.

Solution: a

A-5 Artificial Neural Networks

What happens during *learning* in an artificial neural network?

a) Weights between nodes are adjusted

b) Nodes are added and removed

c) The distribution of training samples is modified

Solution: a

A-6 Support Vector Machine

In a *support vector machine*, what does it mean when a data point has an *alpha*-value of zero?

a) It is impossible to find a solution with the selected kernel

b) The point should not be trusted due to its high variance

c) The point is not used for defining the separating surface

Solution: c

A-7 Ensemble Learning

Which one below best describes the characteristics of Ensemble methods in machine learning?

a) Ensemble learning is not well-suited to parallel computing.

b) The priority is to ensure that the models to be combined perform similarly to each other.

c) Weak learners are trained and combined.

Solution: c

A-8 The Subspace Method

For the subspace methods, a technique of dimentionality reduction is often used to represent the data distribution in each class. Which of these techniques is most suited for this purpose?

a) Principal Component Analysis (PCA).

b) Pulse-Code Modulation (PCM).

c) Phase Change Memory (PCM).

Solution: a

Note: Your answers need be on a solution sheet (**we will not receive this page**).

B Graded problems

A pass is guaranteed with the required points for 'E' below (excluding bonus) in this section *and* the prerequisite in the A-section.

Preliminary number of points required for different grades:

$$\begin{aligned}
22 \leq p \leq 24 &\quad \rightarrow \quad A \\
19 \leq p < 22 &\quad \rightarrow \quad B \\
16 \leq p < 19 &\quad \rightarrow \quad C \\
13 \leq p < 16 &\quad \rightarrow \quad D \\
8 \leq p < 13 &\quad \rightarrow \quad E \\
0 \leq p < 8 &\quad \rightarrow \quad F
\end{aligned}$$

B-1 Terminology (4p)

For each term (a–h) in the left list, find the explanation from the right list which best describes how the term is used in machine learning.

a) k-means

b) A priori probability

c) Curse of dimensionality

d) Negative sample

e) RANSAC

f) Error backpropagation

g) Dimension of a subspace

h) Expectation Maximization

1) Method for estimating the mean of k observations

2) Issues in data sparsity in space

3) Convex optimization

4) Clustering method based on centroids

5) Probability before observation

6) Problems in increasing computation cost

7) The number of base vectors

8) Training samples that are considered as outliers

9) Algorithm to learn with latent variables

10) Algorithm to estimate errors

11) The number of elements in a vector

12) An approach to train artificial neural networks

13) Random strategy for amplitude compensation

14) Training data which is not part of the concept being learned

15) Robust method to fit a model to data with outliers

16) Probability at an earlier time

Solution: a-4, b-5, c-2, d-14, e-15, f-12, g-7, h-9

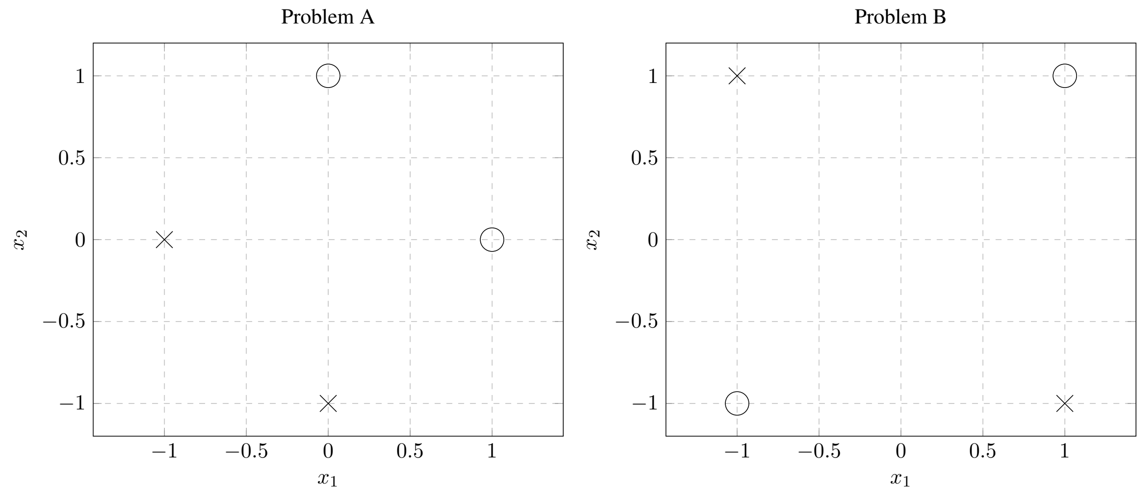

Figure 1. Illustration for Problem B-2

B-2 Probability based learning (3p)

In Figure 1 you see two examples of classification problems where the marker × indicates points from one class and the marker ∘, points from the other class. We call the problem in the *left* plot Problem A and the one on the *right* Problem B. We want to use a Maximum Likelihood classifier based on Gaussian distributions to solve each of these problems. Each class is described by a distribution such that the likelihood of any point x given the class c is:

$$\mathcal{L}(x|c) = \mathcal{N}(x|\mu_c, \Sigma_c) = \frac{1}{2\pi\sqrt{\det(\Sigma_c)}} \exp\left(-\frac{1}{2}(x-\mu_c)^T\Sigma_c^{-1}(x-\mu_c)\right) \qquad \text{with } c \in \{\times, \circ\}.$$

a) For each Problem A and B: what should the mean vectors μ_c be for each class if we consider a generative approach?

b) Consider Problem A. Will the classifier work without errors if we use the mean vectors from question **a** and the identity matrix as covariance for both classes? That is:

$$\Sigma_\times = \Sigma_\circ = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Hint: you do not need to perform complex derivations to answer this question, if you have a convincing theoretical argument.

c) Consider now Problem B. Can you use the same covariance matrices as in question **b** to solve this problem?

Motivate your answers!

NOTE: if you want to refer to the figure, **do not write on this sheet of paper** because we will not receive it with your solutions.

Solution:

a) Because we are considering a generative approach, each distribution will describe the data from a single class and will not be affected by the data from the other class. The distributions that best fit the data will, therefore, have mean vectors that correspond to the sample mean of the points in each class. For Problem A these are:

$$\mu_\times = \begin{bmatrix} -0.5 \\ -0.5 \end{bmatrix} \qquad \mu_\circ = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

For Problem B, both classes have zero mean, that is the mean vectors are the origin for both classes.

b) According to the Maximum Likelihood method, we assign point x to class $\times$ only if

$$\mathcal{L}(x|\times) = \mathcal{N}(x|\mu_\times, \Sigma_\times) > \mathcal{L}(x|\circ) = \mathcal{N}(x|\mu_\circ, \Sigma_\circ).$$

Given the covariance matrix, the distribution for each class will have the same spread in all directions, and the spread will be the same for both classes. The likelihood of a point x given the class will, therefore, depend exclusively on the distance between x and the corresponding mean. In other words, in order for $\mathcal{L}(x|\times)$ to be greater to $\mathcal{L}(x|\circ)$, the point x must be closer to $\mu_\times$ than to $\mu_\circ$. The separating line for the classifier will be the set of points equidistant to $\mu_\times$ and $\mu_\circ$. These points from a line passing through the origin and with slope -1. All the points below the line are classified as $\times$ and all the points above the line as $\circ$, therefore solving the problem without errors.

We can prove the above argument with simple mathematical passages. Considering the definition of Gaussian distribution given in the question, and because $\Sigma_\times = \Sigma_\circ$, the term

$$\frac{1}{2\pi\sqrt{\det(\Sigma_c)}}$$

is the same for both $c = \times$ and $c = \circ$ and can be simplified from the inequality defined above. Furthermore, because the exp function is monotonically increasing, verifying that $\exp(a) > \exp(b)$ is the same as verifying that $a > b$. Finally, we can simplify the term $-\frac{1}{2}$, but, because this is a negative term, we need to invert the direction of the inequality. After all the above passages we obtain:

$$(x - \mu_\times)^T \Sigma_\times^{-1} (x - \mu_\times) < (x - \mu_\circ)^T \Sigma_\circ^{-1} (x - \mu_\circ)$$

Substituting $\Sigma_\times = \Sigma_\circ = I$ we obtain:

$$(x - \mu_\times)^T (x - \mu_\times) < (x - \mu_\circ)^T (x - \mu_\circ)$$

You should already recognize the square Euclidean distance between the point x and each mean vector, but, to make it even more explicit we can expand one of the two terms (omitting the class index for simplicity):

$$[x_1 - \mu_1 \qquad x_2 - \mu_2] \begin{bmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{bmatrix} = (x_1 - \mu_1)^2 + (x_2 - \mu_2)^2$$

As stated before, for a point to be assigned to a certain class, the point has to be closer to that class mean vector.

c) In this case, both the mean vectors and the covariance matrix are equal between the two classes. The distributions will, therefore, not carry any information about the difference between the classes and the problem cannot be solved. Mathematically, for any point x, it will always be $\mathcal{L}(x|\times) = \mathcal{L}(x|\circ)$, which means we are unable to decide between the two classes.

Even if we change the mean vectors, from the discussion made on point **b**, we will only be able to create a linear classifier. On the contrary, the classes in this problems are not linearly separable.

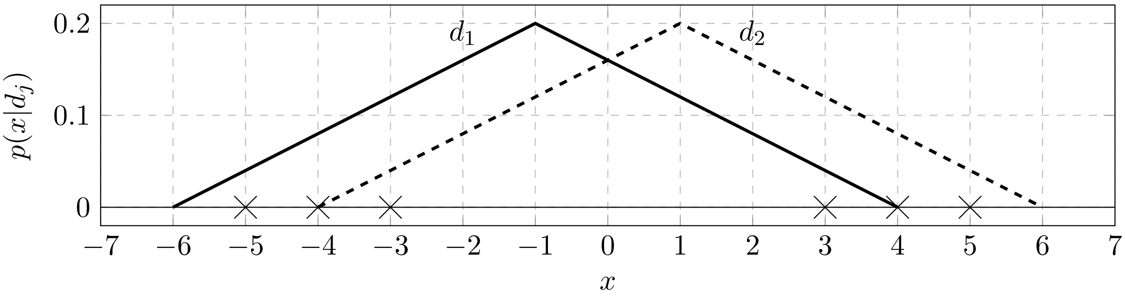

Figure 2. Illustration for Problem B-3

B-3 Probability based Learning: Latent variables (3p)

Figure 2 shows a set of six data points with coordinates $-5, -4, -3, 3, 4, 5$ indicated by the × markers. The figure also shows two probability distribution functions d_1 (thick continuous line) and d_2 (thick dashed line). The PDFs have mean $\mu_1 = -1$ and $\mu_2 = 1$ respectively. They have a maximum value of 0.2 corresponding to the mean, and decrease linearly until they reach zero 5 units away from the mean in both directions.

We do not know which distribution has generated each data point, but we assume the two distribution functions are equally likely *a priori*. We want to use Expectation Maximization to obtain a new estimate of the mean parameter for each distribution, given the data. To do this, perform the following steps:

a) with the help of the plot, evaluate each probability distribution function on each data point, that is, compute $p(x_i|d_j)$. **Hint:** define a symbol $\alpha = \frac{0.2}{5}$ to simplify the calculations throughout this exercise.

b) perform one Expectation step, that is, evaluate the responsibilities $p(d_j|x_i)$ for each point and each distribution.

c) perform one Maximization step, that is, compute the new values for the means of the two distributions. **Hint:** for a set of data points x_i and a set of generic weights w_i, the general formula for the weighted average is:

$$a = \frac{\sum_i w_i x_i}{\sum_i w_i}$$

Solution: Note that the problem is fully symmetric around the origin, so it is enough to get a solution for one of the two distributions and then mirror the solution for the other. For clarity, however, we will compute all the values in this solution.

a) Because the distributions start from the value of 0.2 at the mean and decrease to zero at 5 units from the mean, the slope of the distribution lines, besides a sign, is $\alpha = \frac{0.2}{5}$. This also means that every time we move one unit in x we will increase or decrease $p(x|d_j)$ by α. Using this fact, and with simple geometry, we can compute the distribution functions evaluated at each data point as in the table below:

i	1	2	3	4	5	6
x_i	-5	-4	-3	3	4	5
$p(x_i\|d_1)$	α	2α	3α	α	0	0
$p(x_i\|d_2)$	0	0	α	3α	2α	α

b) The responsibility of each distribution given the point is given by Bayes formula as:

$$p(d_j|x_i) = \frac{p(d_j)p(x_i|d_j)}{p(x_i)} = \frac{p(d_j)p(x_i|d_j)}{p(d_1)p(x_i|d_1) + p(d_2)p(x_i|d_2)},$$

and because we have assumed equal *a priori* probabilities for the distributions, $p(d_1) = p(d_2) = 0.5$, the above formula simplifies to:

$$p(d_j|x_i) = \frac{p(x_i|d_j)}{p(x_i|d_1) + p(x_i|d_2)}.$$

From the table of the previous point, we can obtain the responsibilities as:

i	1	2	3	4	5	6
$p(d_1\|x_i)$	1	1	$\frac{3}{4}$	$\frac{1}{4}$	0	0
$p(d_2\|x_i)$	0	0	$\frac{1}{4}$	$\frac{3}{4}$	1	1

Note that for each data point x_i the responsibilities of the two distributions sum to one.

c) The new values for the means are calculated for each distribution as a weighted average, where the weights are the responsibilities for each point calculated in the previous step, that is:

$$\mu_j = \frac{\sum_i p(d_j|x_i)x_i}{\sum_i p(d_j|x_i)} \tag{1}$$

The sum at the denominator is equal 3 for both distributions, which gives:

$$\mu_1 = \frac{1(-5) + 1(-4) + \frac{3}{4}(-3) + \frac{1}{4}3}{3} = \frac{-9 - \frac{6}{4}}{3} = -\frac{21}{6} = -3.5 \tag{2}$$

For the symmetry we can say that $\mu_2 = 3.5$.

This was not part of the question, but if we take another EM iteration, we see that, because of the shape of the distributions, the new responsibilities will be:

i	1	2	3	4	5	6
$p(d_1\|x_i)$	1	1	1	0	0	0
$p(d_2\|x_i)$	0	0	0	1	1	1

And the new means will just be the average of the three points on the left for d_1 and the average of the three points on the right for d_2, that is $\mu_1 = -4$ and $\mu_2 = 4$. After this, further iterations do not change the solution any longer.

B-4 Classification (3p)

Suppose that we take a data set, divide it into equally-sized training and test sets, and then try out two different classification procedures. First we use the Subspace Method and get an error rate of 12% on the training data. We also get the average error rate (averaged over both test and training data sets) of 20%. Next we use k-nearest neighbor (where $k = 1$) and get an average error rate (averaged over both test and training data sets) of 15%.

a) What was the error rate with the Subspace Method on the test set?

b) Based on these results, indicate the method which we should prefer to use for classification of new observations, with a simple reasoning.

c) Now, if the result of k-nearest neighbor (average error rate being 15%) was based on $k = 5$, would we know if the same conclusion drawn in **b** should still hold? Answer yes or no, with a short reasoning.

Solution:

a) 28%.

b) Subspace Method, because it achieves lower error rate on the test data (28% < 30%); training error for 1-NN is always zero, and therefore the testing error is 30%.

c) No. Because we would not know the error rate for the training with k-NN in the first place, nor for the testing set.

B-5 Decision Forests (2p)

Choose the correct answers in the following questions on Decision Forests.

a) Mainly two kinds of randomness are known to form the principle of Decision Forests. In which two of the following processes are those randomnesses involved?

i. In deciding the number of trees used.
ii. In the rule of terminating a node as a leaf node.
iii. In the way to formulate the information gain.
iv. In generating bootstrap replicas.
v. In feature selection at each node.

b) Suppose we have generated a Decision Forest using five bootstrapped samples from a data set containing three classes, {Green, Blue, Red}. We then applied the forest to a specific test input, x, and observed five estimates of $P(\text{Class is Green}|x)$: 0.3, 0.4, 0.55, 0.6, and 0.75.

There are two common ways to combine these results together into a single class prediction. One is the majority vote approach, and the other is based on the average probability. In this example, what is the final classification under each of these two approaches? Motivate your answer by short phrases.

i. Green in majority vote and Red in averaging.
ii. Blue in majority vote and Green in averaging.
iii. Green in both approaches.
iv. Blue or Red in both approaches.
v. Blue or Red in majority vote and Green in averaging.

Solution:

a)-iv and v, **b**)-iii. The *Green* class has three votes in majority vote, and the average probability, 0.52, is higher than those of other classes.

B-6 Regression with regularization: LASSO (3p)

For a set of N training samples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, each consisting of input vector $\mathbf{x}$ and output y, suppose we estimate the regression coefficients $\mathbf{w}^\top (\in \mathbf{R}^d) = \{w_1, \ldots, w_d\}$ in a linear regression model by minimizing

$$\sum_{n=1}^{N} (y_n - \mathbf{w}^\top \mathbf{x}_n)^2 \quad \text{subject to} \quad \sum_{i=1}^{d} |w_i| \le s$$

for a particular value of s.

For parts **a**) through **c**), indicate which of i. through v. is correct. Briefly justify your answer.

a) As we increase s from 0, the *training* error (residual sum of squares, RSS) will:
i. Steadily increase.
ii. Steadily decrease.
iii. Remain constant.
iv. Increase initially, and then eventually start decreasing in an inverted U shape.
v. Decrease initially, and then eventually start increasing in a U shape.

b) Repeat **a**) for *test* RSS.

c) Repeat **a**) for variance and (squared) bias, respectively.

Solution: When $s = 0$, all w_i are zero; the model is extremely simple, predicting a constant and has no variance with the prediction being far from actual value, thus with high bias. As we increase s, all w_i increase from zero toward their least square estimate values while steadily decreasing the *training* error to the ordinary least square RSS and also decreasing bias as the model continues to better fit training data. The values of w_i then become more dependent on training data, thus increasing the variance. The *test* error initially decreases as well, but eventually start increasing due to overfitting to the training data.

a)-ii.
b)-v.
c)-i for variance, ii for bias.

B-7 Support Vector Machines (3p)

Training a support vector machine using a quadratic kernel

$$\mathcal{K}(\vec{x}, \vec{y}) = (\vec{x}^T \vec{y} + 1)^2$$

has resulted in the following three support vectors:

$$s_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad s_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad s_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

The first (s_1) is a negative sample with $\alpha = \frac{7}{3}$ while the other two are positive samples where $\alpha = \frac{2}{3}$.

a) Determine how the following *new* datapoints will be classified:

$$x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad x_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \qquad x_3 = \begin{bmatrix} -2 \\ 0 \end{bmatrix} \qquad x_4 = \begin{bmatrix} -3 \\ 0 \end{bmatrix}$$

You must show the formulas and calculations used to arrive at your answer.

b) Draw a diagram of the input space with all the relevant points ($s_1, s_2, s_3, x_1, x_2, x_3, x_4$) and show the shape and position of the decision boundary.

Solution:

a) We use the indicator function to classify the new points:

$$\text{ind}(\vec{x}) = \sum_i \alpha_i t_i \mathcal{K}(\vec{x}, \vec{s_i}) = -\frac{7}{3} \cdot \mathcal{K}(\vec{x}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}) + \frac{2}{3} \cdot \mathcal{K}(\vec{x}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}) + \frac{2}{3} \cdot \mathcal{K}(\vec{x}, \begin{bmatrix} 1 \\ 0 \end{bmatrix})$$

Fill in the given values:

$$\text{ind}(x_1) = \text{ind}(\begin{bmatrix} 1 \\ 1 \end{bmatrix}) = -\frac{7}{3} \cdot 1 + \frac{2}{3} \cdot 4 + \frac{2}{3} \cdot 4 = 3$$

x_1 is classified as positive.

$$\text{ind}(x_2) = \text{ind}(\begin{bmatrix} 2 \\ 0 \end{bmatrix}) = -\frac{7}{3} \cdot 1 + \frac{2}{3} \cdot 1 + \frac{2}{3} \cdot 9 = \frac{13}{3}$$

x_2 is classified as positive.

$$\text{ind}(x_3) = \text{ind}(\begin{bmatrix} -2 \\ 0 \end{bmatrix}) = -\frac{7}{3} \cdot 1 + \frac{2}{3} \cdot 1 + \frac{2}{3} \cdot 1 = -1$$

x_3 is classified as negative.

$$\text{ind}(x_4) = \text{ind}(\begin{bmatrix} -3 \\ 0 \end{bmatrix}) = -\frac{7}{3} \cdot 1 + \frac{2}{3} \cdot 1 + \frac{2}{3} \cdot 4 = 1$$

x_4 is classified as positive.

b) A quadratic kernel means that the boundary must be a quadratic curve (ellipse, hyperbola or parabola). In this case it will be a circle where the negative points (s_1 and x_3) should be on the inside and the others on the outside.

B-8 Perceptron Learning (3p)

Consider the training data in the table, where $+$ means a positive sample and $-$ a negative. Show, step by step, the values of the weight vector as it is updated when these samples are used to train a linear classifier using the *perceptron learning rule*. Use the samples in the order as they are listed, and repeat them until all samples are correctly classified.
To make the calculations simple; start by setting all weights to zero, and use a step size (η) of 1. Further, use a threshold function which gives a negative (low) output when the input happens to be *exactly on* the threshold.
Do not only list the successive values of the weights, but also explain what calculations are being done.

1	0	1	0	0	$+$
1	1	1	1	0	$-$
0	1	0	1	0	$+$
1	1	1	0	1	$-$
0	0	1	0	1	$+$
1	1	0	1	1	$-$

Solution:

The weight vector must have six elements (where we interpret the first as the negation of the threshold). The scalar product of each sample (with an extra 1-element) with the weight vector is calculated. If the result is non-positive for a positive sample, the sample is simply added to the corresponding elements in the weight vector. If the scalar product is positive for a negative sample, the sample is subtracted.

Following this recipe, this is how the weights will change as the samples are processed:

Weights						x_0	Sample					Scalar prod.	Result
0	0	0	0	0	0	1	1	0	1	0	0	$0 \leq 0$	Wrong, add sample
1	1	0	1	0	0	1	1	1	1	1	0	$3 > 0$	Wrong, subtract sample
0	0	-1	0	-1	0	1	0	1	0	1	0	$-2 \leq 0$	Wrong, add sample
1	0	0	0	0	0	1	1	1	1	0	1	$1 > 0$	Wrong, subtract sample
0	-1	-1	-1	0	-1	1	0	0	1	0	1	$-2 \leq 0$	Wrong, add sample
1	-1	-1	0	0	0	1	1	1	0	1	1	$-1 \leq 0$	Ok
1	-1	-1	0	0	0	1	1	0	1	0	0	$0 \leq 0$	Wrong, add sample
2	0	-1	1	0	0	1	1	1	1	1	0	$2 > 0$	Wrong, subtract sample
1	-1	-2	0	-1	0	1	0	1	0	1	0	$-2 \leq 0$	Wrong, add sample
2	-1	-1	0	0	0	1	1	1	1	0	1	$0 \leq 0$	Ok
2	-1	-1	0	0	0	1	0	0	1	0	1	$2 > 0$	Ok
2	-1	-1	0	0	0	1	1	1	0	1	1	$0 \leq 0$	Ok
2	-1	-1	0	0	0	1	1	0	1	0	0	$1 > 0$	Ok
2	-1	-1	0	0	0	1	1	1	1	1	0	$0 \leq 0$	Ok
2	-1	-1	0	0	0	1	0	1	0	1	0	$1 > 0$	Ok