

In the name of Allah

Project 5:

Hossein Alizadeh Piralidehi 810100522

Problem considerations:

we are given N iid samples from mixture density:

$$p(x; \lambda) = \sum_{j=1}^k \lambda_j p_j(x)$$

where λ_j is mixture coefficients and $\lambda \in R_+^k$, $1^T \lambda = 1$

and $p_i(x)$ are given densities on R^m

we find ML estimate of λ using convex optimization.

Description:

here we try to calculate Likelihood and formulate problem as convex optimization problem:

$$p(x_1, x_2, \dots, x_N; \lambda) = \prod_{i=1}^N p(x_i; \lambda) = \prod_{i=1}^N \sum_{j=1}^k \lambda_j p_j(x_i)$$
$$\ell = \log(p(x_1, x_2, \dots, x_N; \lambda)) = \sum_{i=1}^N \log \left(\sum_{j=1}^k \lambda_j p_j(x_i) \right)$$

now considering probability densities $p_i(x)$ are known we can rewrite log-likelihood:

$$c_i^T \lambda = \left(\sum_{j=1}^k \lambda_j p_j(x_i) \right) \quad \text{which } j\text{th element of } c_i \text{ is : } p_j(x_i)$$
$$\ell = \log(p(x_1, x_2, \dots, x_N; \lambda)) = \sum_{i=1}^N \log(c_i^T \lambda)$$

$c_i^T \lambda$ is affine in λ , and $\log(x)$ is concave function so $\log(c_i^T \lambda)$ which is combination of concave and affine will be concave, as we know summation will preserve concavity so log-likelihood function is concave in λ .

finally optimization problem can be described as below:

$$\max \sum_{i=1}^N \log(c_i^T \lambda)$$

s. t.

$$1^T \lambda = 1$$
$$\lambda \geq 0$$

which is an convex optimization problem with affine constraints.

Simulation with given parameters:

Part a)

load parameters

```
clc;clear;close all;

lambdas_true = [0.3; 0.5; 0.2];
load('mixture_coeffs_data.mat');

C = densities;
```

solve convex problem using cvx toolbox

```
cvx_begin
    variables lambdas(size(densities,2));
    ML_Objective = 0;
    for i = 1 : size(densities,1)
        ML_Objective = ML_Objective + log(C(i,:)*lambdas);
    end
    maximize( ML_Objective );
    subject to
        ones(1,size(densities, 2))*lambdas == 1;
        lambdas >= 0;
cvx_end
```

Successive approximation method to be employed.

For improved efficiency, SDPT3 is solving the dual problem.

SDPT3 will be called several times to refine the solution.

Original size: 304 variables, 103 equality constraints

100 exponentials add 800 variables, 500 equality constraints

Cones Mov/Act	Errors			Status
	Centering	Exp cone	Poly cone	
100/100	8.000e+00	4.548e+00	0.000e+00	Solved
100/100	2.152e+00	3.041e-01	0.000e+00	Solved
100/100	1.046e-01	7.037e-04	0.000e+00	Solved
100/100	1.679e-02	1.774e-05	0.000e+00	Solved
100/100	2.234e-03	3.137e-07	0.000e+00	Solved
0/100	3.012e-04	5.504e-09	0.000e+00	Solved

Status: Solved

Optimal value (cvx_optval): -206.882

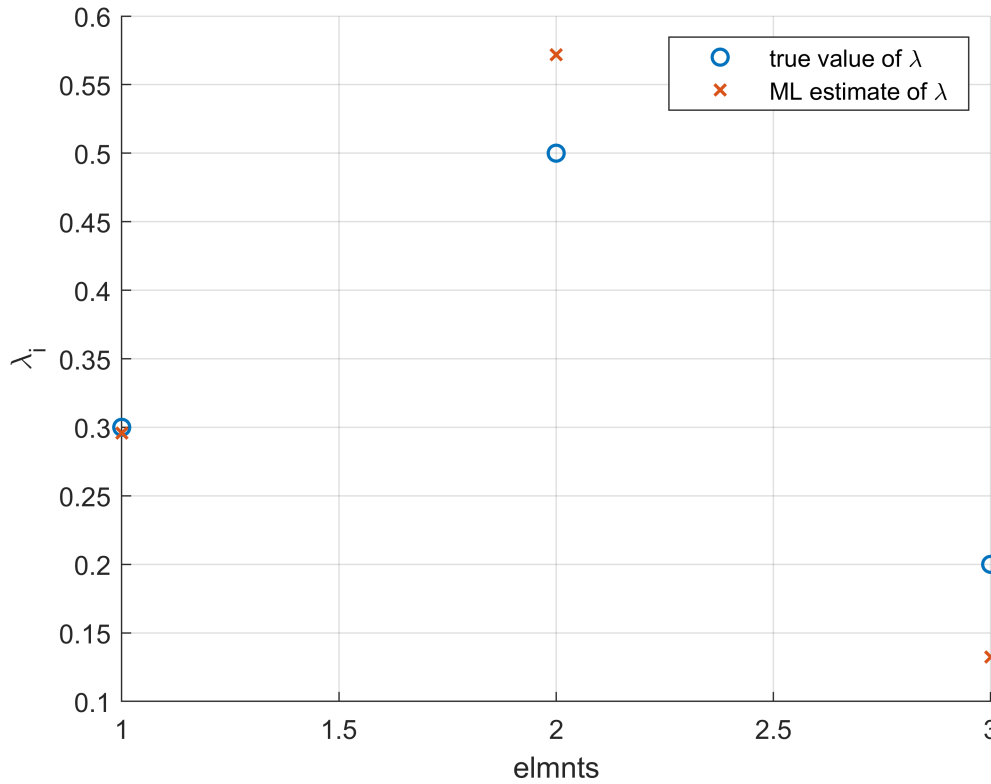
plot outputs

```
figure
hold on
```

```

grid on
plot(lambdas_true,'o','LineWidth',1.2)
plot(lambdas, 'x','LineWidth',1.2)
xlabel('elmnts')
ylabel('\lambda_i')
legend('true value of \lambda', 'ML estimate of \lambda')

```



as we see for $N = 100$ samples of mixture density, ML estimate of lambdas are close to true value of them. and by increasing number of samples this estimation will be more accurate and converge to true value of lambdas.