In a search of router for mixture of experts

V. O. Miroshnychenko[1], **T. B. Skorobohach**[2]

[1]Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

vitaliy.miroshnychenko@knu.ua

[2]NTU Sikorskiy, Kyiv, Ukraine

tetiana.skorobohach@ntu.ua

Introdution

Mixture models have long been a fundamental tool in statistical analysis for modeling heterogeneous data, dating back to the work of Karl Pearson in 1894 [1], who applied a mixture of normal distributions to study bimodal data. Pearson's approach laid the foundation for later developments in the theory of mixtures of probability distributions. Over time, a variety of mixture models have been proposed, including Gaussian Mixture Models (GMMs), introduced by Titterington et al. (1985) [2], and extended to other distributions such as Poisson and exponential mixtures, enabling the modeling of diverse real-world phenomena.

Mixture models are probabilistic models that assume that the data is generated from a combination of several distributions, each representing a different subpopulation. The probability density function (pdf) of a mixture model can be written as a weighted sum of component distributions:

$f(x|\theta) = \sum_{k=1}^{M} \pi_k f_k(x|\theta_k)$

where:

1 M is the number of components,

2 π_k are the mixing coefficients, such that $\sum_{k=1}^{M} \pi_k = 1$ and $\pi_k \geqslant 0$,

3 $f_k(x|\theta_k)$ is the probability density function of the k-th component, with parameters θ_k,

4 $\Theta = \{\theta_1, ..., \theta_M\}$ denotes the set of parameters for all the components.

These models have become widely used for clustering, classification, and density estimation, as they provide a flexible framework for capturing the inherent structure

in complex datasets. The Expectation-Maximization (EM) algorithm, proposed by Dempster et al. (1977) [3], is commonly employed to estimate the parameters of mixture models in a maximum likelihood framework.

Mixture of experts

The Mixture of Experts (MoE) model is an advanced extension of the mixture model framework, introduced by Steven J., Nowlan and Geoffrey E. Hinton [5] (1990) and Jacobs, Hinton et al. [4] (1991). The core idea is to partition the input space and assign specialized "expert" models to different regions, while a gating network determines the contribution of each expert based on the input. The MoE model elegantly combines both model selection and model combination techniques, offering a powerful and flexible framework for solving complex prediction tasks.

Unlike standard mixture models, which focus on probabilistic combinations of distributions, the MoE approach uses multiple models (or experts) that focus on specific subregions of the data, as directed by the gating network. This method has been widely used in areas such as regression, classification, and time-series forecasting, providing significant improvements over traditional single-model approaches in handling heterogeneous or multi-modal data.

The general form of a Mixture of Experts model can be expressed as:

$$y(x|\theta) = \sum_{k=1}^{M} g_k(x|\phi) h_k(x|\theta_k)$$

or

$$y(x|\theta) = argmax_{k=1,\bar{M}} g_k(x|\phi) h_k(x|\theta_k)$$

1 M is the number of components,

2 $g_k(x|\phi)$ is a gating algorithm (router), which assigns probabilities to each expert based on the input x with parameters ϕ,

3 $h_k(x|\theta_k)$ is the output of the k-th expert models, with its own set of parameters θ_k,

4 $\Theta = \{\theta_1, ..., \theta_M\}$ denotes the set of parameters for all the components.

The gating function $g_k(x|\phi)$ satisfies the constraints

$$\sum_{k=1}^{M} g_k(x|\phi) = 1 \text{ and } g_k(x|\phi) \geqslant 0.$$

This formulation allows MoE to dynamically assign responsibility to different expert models depending on the input, thus optimizing performance for a wide range of tasks.

Since its introduction, MoE has been extended and widely applied in neural networks, ensemble learning, and more recently in deep learning frameworks, offering scalability and flexibility in high-dimensional and large-scale data contexts.

Mixture of varying concentrations

Lets use mixture of varying concentrations from Maiboroda, Sugakova, (2008) [6]. Let the O be a research object. This object has unobservable characteristics $\kappa(O) \in \{1, ..., M\}$. $\kappa(O)$ is a number of component (population) for which the O belongs. Also, the object O has observable characteristics. For it we knows probabilites to belong for a fixed component $\mathbb{P}\{\kappa(O) = k\}$. Additionaly for O we knows characteristics (regressors/features) $\xi(O) = \left(\xi^1(O), ..., \xi^D(O)\right)^T$. The distribution of $\xi(O)$ is dependent of $\kappa(O)$ (unobserve component number). By the F_m, $1 \leqslant m \leqslant M$ we denote a features distribution from the m-th component:

$$F_m(A) = P\Big(\xi(O) \in A | \kappa(O) = m\Big),$$

for anu measurable $A \subset \mathbb{R}^D$.

Consider objects $O_{j;n}$, $j = 1, ..., n$ from different components. Those object belongs to different components with known probabilites $p^k_{j;n} = P\{\kappa(O_{j;n}) = k\}$, $\kappa_j = \kappa(O_{j;n})$. We denote a probability of the object j to belong to component k by a symbol $p^k_{j;n}$.

Let Ξ_n be a sample of range n:

$$\Xi_n = \left(\xi(O_{j;n})\right)_{j=1}^{n} = \left(\xi_{j;n}\right)_{j=1}^{n}.$$

We assume that random variables $(\xi_{j;n}, \kappa_j)$ are independent for different j. By the P_n we denote a matrix of mixting probabilites

$$P_n = \left(p^k_{j;n}\right)_{j=1,k=1}^{n,M}.$$

For different $j = 1, ..., n$, $\xi_{j;n}$ have the next distribution

$$\mathbb{P}\{\xi_{j;n} \in A\} = \sum_{m=1}^{M} p_{j;n}^{m} F_m(A),$$

for any measurable $A \subset \mathbb{R}^D$.

Minimax measures and estimates

For F_m we consider nonparametric model. Weighted empirical distribution function (w.e.d.f) with weights $a_{j;n}$ is called function

$$\hat{F}_n(x, a) := \sum_{j=1}^{n} a_{j;n} I[\xi_{j;n} < x].$$

The example of (w.e.d.f) is an estimators for distribution F_m with weights $a^m = (a_{1;n}^m, ..., a_{n;n}^m)^T$

$$\hat{F}_n^{(m)}(x, a) := \sum_{j=1}^{n} a_{j;n}^m I[\xi_{j;n} < x], \tag{1}$$

where weights $a_{j;n}^m$ are defined by the next equation

$$A_n = \left(a_{j;n}^k\right)_{j=1,k=1}^{n,M} = P_n^T (\Gamma_n)^{-1}. \tag{2}$$

Here Γ_n is a Gram matrix for vectors from P_n

$$\Gamma_n = P_n P_n^T = (\langle p^k, p^l \rangle)_{k,l=1}^{M}.$$

It is easy to show that the next equation holds

$$P_n A_n = \left(\sum_{i=1}^{n} a_{i:n}^k p_{i:n}^t \right)_{t,k=1}^{M} = I_M.$$

The article [8] shows that $\hat{F}_n^{(m)}$ is a minimax estimator of $F^{(m)}$. Generally speaking, the function $\hat{F}_n^{(m)}$ from (1) is not a CDF. There are always negative weights from $a_{j;n}^m$, so the function $\hat{F}_n^{(m)}(x, a)$ is not non-decreasing (the estimator for variance might be negative). Maiboroda [6] (paragraph 2.3) shows how to build enhansed weights $a_{j;n}^k$ without negative values and then it is possible to use them to build estimators for non-negative values.

In article R. Maiboroda, O. Sugakova [6] described theorem about minimax moment estimator aymptotic properties. Further this this topic is researched in [12] R. Maiboroda, O. Sugakova, and A. Doronin. Authors builds distribution parameters using minimax weights. In this article we use the next theorem for moment estimation in our case.

Consider a function $g : \mathbb{R}^D \to \mathbb{R}$. Then the k-th functional moment for

$$\overline{g}_k = \int g(x)F_k(dx) = \mathbb{E}\, g(\xi_\cdot^{(k)})$$

can be estimated by

$$\hat{g}_{k:n} = \int g(x)\hat{F}_k(dx) = \sum_{j=1}^{n} a_{j:n}^k g(\xi_j).$$

Here

$$\hat{F}_k(A) = \sum_{j=1}^{n} a_{j:n}^k I[\xi_i \in A]$$

is a (w.e.d.f) estimator for $F_k(A)$.

Proposition 0.1. *Assume that for any* $k = 1, ..., M$

1 $\bar{g}_k$ *exists.*

2 $\sup_{j,n} |a_{j:n}^k| < \infty$.

Then $\hat{g}_{k:n} \xrightarrow{\mathbb{P}} \overline{g}_k,\ n \to \infty$

Generally speaking, the weights $a_{i:n}^k$ are not bounded. For the case, when $p_1^1 = \frac{1}{n}$ and $p_1^2 = 1 - \frac{1}{n}$ and $p_{i:n}^k = k - 1$ (0 or 1) for $2 \leqslant i \leqslant n$, the $a_{1:n}^1$ rises infinitly together with n:

$$a_{1:n}^1 = \frac{1}{n}\Big(\frac{n^2}{n-1}\big((1-\frac{1}{n})^2 + n - 1\big)\Big) - (1-\frac{1}{n})\Big(\frac{n^2}{n-1}\frac{1}{n}(1-\frac{1}{n})\Big)$$

The second term goes to unit and the first is equivalent to n, as $n \to \infty$. So $a_{1:n}^1/n \to 1$, as $n \to \infty$. This example shows if there a component with fixed number of object, then their weight are increasing together with n.

R. Maiboroda and O. Sugakova in [8] describeed asymptotic properties for minimax weights $a_{i:n}^k$. The next proposition tells about that properties.

Proposition 0.2. *If the condition* $\det \lim_{n\to\infty} \frac{1}{n}\Gamma_n > 0$ *holds, then:*

$$\sup_{\substack{j=\overline{1,n,}\\ k=\overline{1,M}}} |a^k_{j:n}| = O(n^{-1}) \text{ as } n \to \infty.$$

In this article we will prove analog of this theorem for parametric router function $p(x, \theta)$.

Minimax router for mean estimation

In article [9] we describe Least Square (LS) and Smoothed empirical likelihood estimator parameter estimators for router function (with parametric assumtion for router). For the LS estimator the consistency is proven. Here I would like to describe one more estimator that is based on the same ideas as K-means clusterizer.

The K-means clusterizer is a next approach. Lets denote the feature space $\mathbb{R}^d$ split as $S = (S_1, ..., S_M)$ where each S_k is called a cluster with a centroid μ_k. The goal is to reach the minimum of total variance:

$$S = argmin_S \sum_{k=1}^{M} \sum_{x\in S_k} ||x - \mu_k||^2.$$

Generally, the problem is NP-hard, but there are a lot of heuristics to solve it. Lloyd's algorithm [14] or Voronoi iterations are the simpliets approaches.

We would like to start from the $d = 1$. Our idea is to substitute the term $\sum_{x\in S_k} ||x - \mu_k||^2$ by the consistent variance estimator with minimax weights $\sum_{i=1}^n a^k_{i:n}(X_i - a)^2$. In this approach $a^k_{i:n}$ are dependent on parameters θ and we make assumption about $p^k_{i:n} = p^k_{i:n}(\theta)$.

Consider a sample from the mixture of varying concentrations $X = (X_1, ..., X_n)$, $X_i \in R^1$, and matrix of mixing probabilities $P_n = (p^1_{:n}, ..., p^M_{:n})$, $p^k_{:n} = (p^k_{1:n}, p^k_{2:n}, ..., p^k_{n:n})$ and $\sum_{k=1}^M p^k_{i:n} = 1$, for each $1 \leqslant i \leqslant n$, $0 \leqslant p^k_{i:n} \leqslant 1$. Also we assume that X_i are not all the same. Consider minimax weights $A_n = P_n(P_n^T P_n)^{-1}$.

Let's denote F_k a CDF for component k, $1 \leqslant k \leqslant M$. This CDF is known till the mean vector $\mu_k = E[X|\kappa(X) = k] \in R^D$.

Under the propostion 0.1 assumptions, the estimator $\hat{\mu}^a_k = \sum_{i=1}^n a^k_{i:n} X_i$ is a non-biased and consistent estimator.

By the method of monents, we can write the loss function $L_k(a) = \sum_{i=1}^n a_{i:n}^k (X_i - a)^2$, for $a \in \Theta_\mu \subseteq \mathbb{R}$. By differentiation we observe that estimator for a is $\hat{a} = \sum_{i=1}^n a_{i:n}^k X_i$ is point of extremum. We should denote that this loss function is not bounded – some weights are negative. This means some iterative optimization approcahes like gradient descent or Newton-Raphson might fail to find local point of minima.

Now we consider the wider model with the next matrix $P_n = P_n(\theta)$ is dependent on parameter vector $\theta \in \Theta_r \subset R^T$, where $p_{i:n}(\theta) = (p_{i:n}^1(\theta), ..., p_{i:n}^M(\theta)) = p(X_i, \theta)$ we will call a router. We consider paramtric assumption for the router so the function p is known. Then, the matrix A_n is dependent on θ too.

Generally speaking, router function P_n could be dependent on parameter $\mu \in \Theta_\mu^M$, $P_n = P_n(\theta, \mu)$ as well, but we do not consider this case in this article.

We will build routers for μ_k estimation by total mean variance minimization. Lets denote

$$L(\theta) = \Big| \sum_{k=1}^M \sum_{i=1}^n a_{i:n}^k (X_i - \hat{\mu}_k)^2 \Big|.$$

This function is hardly to minimize using classic derivative approach. We take the absolute value because there are some negative weights and they are not bounded as shown before. One more issue is in taking derivative $\frac{\partial A_n(\theta)^{-1}}{\partial \theta}$, especially when number of components $M > 4$. Even block matrix inversion from [13] is not easy to implement and not all popular framworks (like Tensorflow) implements inverted matrix differentiation. To bypass these issues we build the upper bound and will minimize the bound.

In this paper we consider several functions as a router. The first one is logistic

$$p_{i:n}^1 = p_{i:n}^1(\theta) = \frac{1}{1 + e^{-\langle X_i, \theta \rangle}}, \text{ and } p_{i:n}^2 = 1 - p_{i:n}^1,$$

and two more based on simlpe functions (those look like decision trees and)

It is easy to show using aritmetical properties over continuous functions that $\det(\Gamma_n(\theta)) \to 0$ as $||\theta|| \to 0$ and, as a result, $\sup\limits_{\substack{1 \leqslant i \leqslant n, \\ 1 \leqslant k \leqslant M}} |a_{i:n}^k| \to \infty$.

The properties of minimax weigths are dependent of the chsen router function. So for applications we need to have the analog of a proposition 0.2. In our case $A_n = A_n(\theta)$ and we would like to find conditions for equality

$$\sup_{\substack{\theta \in \Theta_r, i = \bar{1,n}, \\ k = \bar{1,M}}} |a_{i:n}^k(\theta)| = O(n^{-1}) \tag{3}$$

Theorem 0.1. *Lets denote* $P_n(\theta)P_n^T(\theta) = \Gamma_n(\theta)$ *such that next conditions holds,*

1 $\frac{1}{n}\Gamma_n(\theta) \to \Gamma_\infty(\theta)$, *as* $n \to \infty$ *for any* $\theta \in \Theta_r$,

2 $\sup_{\theta\in\Theta_r} ||\Gamma_\infty^{-1}(\theta)|| < \infty$,

Then the equation (3) holds.

Proof. Lets fix $\tau \in \Theta_r$. From the first condition and continuous transformations, we have $\forall\varepsilon > 0,\ \exists N : \forall n > N$:

$$0 \leqslant n||\Gamma_n^{-1}(\tau)|| < \varepsilon + ||\Gamma_\infty^{-1}(\tau)|| \leqslant \varepsilon + \sup_{\theta\in\Theta_r} ||\Gamma_\infty^{-1}(\theta)|| = c_\varepsilon$$

This means, $\sup_{\tau\in\Theta_r} n||\Gamma_n^{-1}(\tau)|| \leqslant c_\varepsilon$ and

$$\sup_{\tau\in\Theta_r} ||\Gamma_n^{-1}(\tau)|| = O(\frac{1}{n}), \tag{4}$$

as $n \to \infty$ (big O notation).

Then, from (4) and $a_{i:n}^k(\theta)$ definition and auchy–Schwarz enequality:

$$\sup_{\substack{\theta\in\Theta_r, i=\overline{1,n},\\ k=\overline{1,M}}} |a_{i:n}^k(\theta)| \leqslant \sup_{\substack{\theta\in\Theta_r, i=\overline{1,n},\\ k=\overline{1,M}}} \sum_{t=1}^{M} p_{i:n}^t(\theta)(\Gamma_n^{-1}(\theta))_{t,k} \leqslant M \sup_{\theta\in\Theta_r} ||\Gamma_n^{-1}(\theta)|| = O(\frac{1}{n}),$$

as $n \to \infty$.

□

Lets consider enequality for $L(\theta, \nu)$. In the next theorems we will show that

$$L(\theta,\nu) \leqslant \sqrt{\sum_{k=1}^{M} \frac{1}{\lambda_k^2(\theta)}} \sqrt{\sum_{k=1}^{M}\sum_{s=1}^{M}(\sum_{i=1}^{n} p_{i:n}^s(\theta)(X_i - \nu_k)^2)^2} = \tilde{Q}(\theta,\nu) \tag{5}$$

and for M=2

$$L(\theta,\nu) \leqslant \sqrt{\sum_{k=1}^{M} \frac{1}{\lambda_k^2(\theta)}} \sqrt{\sum_{k=1}^{M}(\sum_{i=1}^{n} p_{i:n}^k(\theta)(X_i - \nu_k)^2)^2} = Q(\theta,\nu) \tag{6}$$

where $\lambda_k(\theta)$ are eigenvalues of matrix $\Gamma_n = (P_n^T P_n)$.

The last enequality is true if Γ_n^{-1} is a Stieltjes matrix (if it has non-positive off-diagonal elements, z-matrix). For $M = 2$, Γ_n^{-1} is always a Stieltjes matrix. For $M \geqslant 3$ the matrix $\Gamma_n(\theta)^{-1}$ might contains positive elements.

More information you can find at chapter 3, Theorem 2.3 from [10]. 50 equivalent definitions for m-matrix are listed there.

Lets prove the equality (5).

Theorem 0.2. *For matrix of weights A_n and matrix of mixing probabilities P_n, for any $\theta \in \Theta_r$ and for $s(x, v) \geqslant 0$, $v_{(k)} \in \Theta_\mu$, $1 \leqslant k \leqslant M$ and for non-singular $\Gamma_n(\theta)$, $\theta \in \Theta_r$ and $v \in \Theta_\mu$ the next enequality holds:*

$$L(\theta, v) \leqslant \tilde{Q}(\theta, v)$$

Proof. By the definition of $a_{j;n}^k$,

$$L(\theta, v) = \sum_{k=1}^{M} \sum_{j=1}^{n} (\sum_{s=1}^{M} p_{j;n}^s(\theta)(\Gamma_n(\theta)^{-1})_{s,k}) s(\xi_j, v_{(k)}) =$$

$$= \sum_{k=1}^{M} \sum_{s=1}^{M} [\Gamma_n(\theta)^{-1})_{s,k}] [\sum_{j=1}^{n} (p_{j;n}^s(\theta) s(\xi_j, v_{(k)})]$$

by the Cauchy–Schwarz inequality

$$L(\theta) \leqslant \tilde{Q}(\theta) = ||\Gamma_n^{-1}(\theta)||_F \sqrt{\sum_{k=1}^{M} \sum_{s=1}^{M} [\sum_{j=1}^{n} (p_{j;n}^s(\theta) s(\xi_j, v_{(k)})]^2} \tag{7}$$

The matrix Γ_n is symmetric, so

$$||\Gamma_n^{-1}(\theta)||_F^2 = \mathrm{Tr}(\Gamma_n^{-1}(\Gamma_n^{-1})^T) = \mathrm{Tr}(\Gamma_n^{-2}) = \sum_{m=1}^{M} \frac{1}{\lambda_m(\theta)^2},$$

where $(\lambda_m(\theta), m = 1, ..., M)$ are eigenvalues of $\Gamma_n(\theta)$, and they are dependent on θ.

□

The enequality (7) is not the best ofr practical usage. More interesting result were obtained by mistake from $\tilde{Q}(\theta)$ where all indexes s were replaced by k and only one sum left.

$$Q(\theta, v) = ||\Gamma_n^{-1}(\theta)||_F \sqrt{\sum_{k=1}^{M} [\sum_{j=1}^{n} (p_{j;n}^k(\theta) s(\xi_j, v_{(k)})]^2} \leqslant \tilde{Q}(\theta, v) \tag{8}$$

By the mistake initially we firstly implemented router using loss function $Q(\theta, \nu)$ and obtained nteresting results for regression problem. When we tried $\tilde{Q}(\theta, \nu)$ is was not the best case. The next theorem will show that at least for $M = 2$ (and some other cases), $L(\theta, \nu) \leqslant Q(\theta, \nu) \leqslant \tilde{Q}(\theta, \nu)$

Theorem 0.3. *1. If for any $z \in \mathbb{R}^D$, $\theta \in \mathbb{R}^d$, $s(z, \theta) \geqslant 0$ then for $M = 2$:*

$$L(\theta, \nu) \leqslant Q(\theta, \nu).$$

2. If for any $z \in \mathbb{R}^D$, $\theta \in \mathbb{R}^d$ $s(z, \theta) \geqslant 0$ and $(\Gamma_n^{-1})_{k,s} \leqslant 0$ for $1 \leqslant k \neq s \leqslant M$ (the matrix Γ_n^{-1} is a Stieltjes matrix), than for any $M > 2$ and $\forall\theta$ the next enequalty holds

$$L(\theta, \nu) \leqslant Q(\theta, \nu).$$

Proof. For any fixed $\theta \in \Theta_r$ and $\nu = (\nu^{(1)}, ..., \nu^{(M)})$

$$L(\theta, \nu) = \sum_{k=1}^{M} \sum_{j=1}^{n} (\sum_{s=1}^{M} p_{j;n}^{s}(\theta)(\Gamma_n(\theta)^{-1})_{s,k}) s(\xi_j, \nu_{(k)}) =$$

$$= \sum_{k=1}^{M} \sum_{s=1}^{M} [\Gamma_n(\theta)^{-1})_{s,k}] [\sum_{j=1}^{n} (p_{j;n}^{s}(\theta) s(\xi_j, \nu_{(k)})] =$$

$$\sum_{k=1}^{M} [\Gamma_n(\theta)^{-1})_{k,k}] [\sum_{j=1}^{n} (p_{j;n}^{k}(\theta) s(\xi_j, \nu_{(k)})] + \sum_{k=1}^{M} \sum_{s=1, s\neq k}^{M} [\Gamma_n(\theta)^{-1})_{s,k}] [\sum_{j=1}^{n} (p_{j;n}^{s}(\theta) s(\xi_j, \nu_{(k)})] \leqslant$$

$$Q(\theta, \nu) + \sum_{k=1}^{M} \sum_{s=1, s\neq k}^{M} [\Gamma_n(\theta)^{-1})_{s,k}] [\sum_{j=1}^{n} (p_{j;n}^{s}(\theta) s(\xi_j, \nu_{(k)})].$$

For $M = 2$ the second part is always negative. By the condition of the theorem $\sum_{j=1}^{n} p_{j;n}^{1}(\theta) s(\xi_j, \nu_{(2)})$ is always non-negative. Lest show that number $(\Gamma_n(\theta)^{-1})_{1,2}$ is negative or zero. Easy to show that

$$\left(\Gamma_n(\theta)^{-1}\right)_{1,2} = -\frac{1}{\det(\Gamma_n)} \Gamma_n(\theta)_{1,2},$$

and $\det(\Gamma_n) > 0$ because Γ_n is a symmetric, positive and positive-definite matrix.

In a general case from the assumption $(\Gamma_n^{-1})_{k,s} \leqslant 0$ for $1 \leqslant k \neq s \leqslant M$ we have an equality.

□

For statistical inference and machine learning applications the loss $Q(\theta, \nu)$ is more preferable than $\tilde{Q}(\theta, \nu)$. The first loss function is more suitable because in this case we should not to minimize expert's quality on other expert's regions.

Mean estimators

Previously we use $\hat{\mu}_k^a$ to estimate router's paramaters θ. Lets denote matrix $\mu = (\mu_1, ..., \mu_M)$. Now we consider enequality ($For\ M = 2$)

$$L(\theta, \mu) = \Big| \sum_{k=1}^{M} \sum_{i=1}^{n} a_{i:n}^k (X_i - \mu_k)^2 \Big| \leqslant$$

$$\leqslant \sqrt{\sum_{k=1}^{M} \frac{1}{\lambda_k^2(\theta)}} \sqrt{\sum_{k=1}^{M} (\sum_{i=1}^{n} p_{i:n}^k(\theta)(X_i - \mu_k)^2)^2} = Q(\theta, \mu)$$

So the μ and θ estimators are $\hat{\mu}^p = (\hat{\mu}_1^p, ...\hat{\mu}_M^p)$, $\hat{\theta}^p$, $\hat{\mu}^a = (\hat{\mu}_1^a, ..., \hat{\mu}_M^a)$ and $\hat{\theta}^a$.s

$\hat{\mu}^a$ estimator. Consider estimates

$$\hat{\mu}_k^a = \sum_{i=1}^{n} a_{i:n}^k X_i, \text{ for } 1 \leqslant k \leqslant M$$

$$\hat{\theta}^a = argmin_{(\mu,\theta)} L(\theta, \hat{\mu}^a)$$

$\hat{\mu}^p$ estimator. Consider estimates

$$(\hat{\mu}^p, \hat{\theta}^p) = argmin_{(\mu,\theta)} Q(\theta, \mu)$$

The minimum point exists because the function Q is countinuous by μ_k and goes to infinity when the μ norm goes to infinity. Moreover, here we wil find this estimator.

For the loss function and fixed θ lets find minimum by μ of $Q(\theta, \mu)^2$

$$Q(\theta, \mu)^2 = \sum_{k=1}^{M} \frac{1}{\lambda_k^2(\theta)} \sum_{k=1}^{M} (\sum_{i=1}^{n} p_{i:n}^k(\theta)(X_i - \mu_k)^2)^2$$

So, find the solution of the equstion $\nabla Q(\theta, \mu)^2$:

$$\nabla_{\mu_l} Q(\theta, \mu)^2 = \sum_{k=1}^{M} 2(\sum_{i=1}^{n} p_{i:n}^k (X_i - \mu_k)^2) \sum_{i=1}^{n} p_{i:n}^k (-2)(X_i - \mu_k)(\nabla_{\mu_l} \mu_k) =$$

$$-4(\sum_{i=1}^{n} p_{i:n}^{l}(X_i - \mu_l)^2) \sum_{i=1}^{n} p_{i:n}^{l}(X_i - \mu_l) = 0$$

The first term $\sum_{i=1}^{n} p_{i:n}^{k}(X_i - \mu_k)^2$ is non-negative. There is enough to solve

$$\sum_{i=1}^{n} p_{i:n}^{k}(X_i - \mu_k) = 0.$$

As a result we have something that looks like Nadaraya-Watson formula and mean estimator for Gaussian mixture model from [11],

$$\hat{\mu}_k^p = \sum_{i=1}^{n} X_i p_{i:n}^{k} / \sum_{i=1}^{n} p_{i:n}^{k}.$$

The second derivetive is

$$\frac{\partial^2 Q(\theta, \mu)}{\partial \mu_k^2} = 8\Big(\sum_{i=1}^{n} p_{i:n}^{k}(X_i - \mu_k)\Big)^2 + 4\sum_{i=1}^{n} p_{i:n}^{k}(X_i - \mu_k)^2 \sum_{i=1}^{n} p_{i:n}^{k} \geqslant 0$$

The second term is always greater than zero (exept the singular case when all $p_{i:n}^{k}$ equals to zero. As a result,

$$\frac{\partial^2 Q(\theta, \mu)}{\partial \mu_k^2}\Big|_{\mu_k = \hat{\mu}_k^p} = 4\sum_{i=1}^{n} p_{i:n}^{k}(X_i - \hat{\mu}_k^p)^2 \sum_{i=1}^{n} p_{i:n}^{k} > 0$$

This shows that $\hat{\mu}_k^p$ is a point of minima. It means that for a fixed estimator $\hat{\mu}^p$ we can just minimize Q by θ.

Note. There are case when $\hat{\mu}^p = \hat{\mu}^a$. It is true when router probabilities are descrete, so $p_{i:n}^{k} \in \{0, 1\}$.

Router estimation

From previouse sections and theorem 0.3 it is obviouse that next enequalities are hold,

$$L(\theta, \hat{\mu}^a) \leqslant L(\theta, \hat{\mu}^p) \leqslant Q(\theta, \hat{\mu}^p) \leqslant Q(\theta, \hat{\mu}^a)$$

These enequalities will help us to fit the parameteres of the router. As we said before, the functions $L(\theta, \hat{\mu}^a), L(\theta, \hat{\mu}^p)$ and $Q(\theta, \hat{\mu}^a)$ are bad choise to fit router. All fo them has problems with derivetives and first two also might not have lower bound (it is dependent on router chose). Later we will show for some cases there are might be

problems with gram matrix Γ_n singularity. As a result, the loss function $Q(\theta, \hat{\mu}^p)$ is a good candidate to build router. Even the fact that we should take the derivative of eigen value by the parameter is not a problem and some frameworks can do this work for us (like Tensorflow).

So the estimator $\hat{\mu}^a$ is not the best for Q minimization. We have not found any formulas for router's parameters using different assumptions yeat. So when the estimator for μ is fixed we propose to mimimize L and Q losses by the θ (and evaluate their gradients) using libraries like Tensorflow or SciPy.

Decision trunc router

We will start from the simple decision trunc router with $M = 2$. It means, $p^2_{i:n} = I[X_i > \theta]$ and $p^1_{i:n} = 1 - p^2_{i:n}$. This function is not differentiable but it completely replace all linear routers (logistic, LDA, kmeans, mixture model, etc). For $D = 1$ case we should just select only this threshold θ.

Needed to say, that for this router estimators $\hat{\mu}^p_k$ and $\hat{\mu}^a_k$ are completly identical for any fixed $\theta \in \Theta_r$.

By the definition, $a^k_{i:n} = \sum_{s=1}^{M} p^s_{i:n}(\Gamma_n^{-1})_{s,k}$. Lets notice that $(\Gamma_n)_{k,s} = \langle p^s_\cdot, p^k_\cdot \rangle = 0$ if $k \neq s$. As a result, $(\Gamma_n^{-1})_{k,s} = 0$. So

$$a^k_{i:n} = p^k_{i:n}(\Gamma_n^{-1})_{k,k} = p^k_{i:n} / \sum_{i=1}^{n} (p^k_{i:n})^2 = p^k_{i:n} / \sum_{i=1}^{n} p^k_{i:n}.$$

We have

$$\hat{\mu}^p_k = \sum_{i=1}^{n} X_i p^k_{i:n} / \sum_{i=1}^{n} p^k_{i:n} = \sum_{i=1}^{n} X_i a^k_{i:n} = \hat{\mu}^a_k.$$

In this case we can try different θ from the set $\{\frac{X_{(k)}+X_{(k+1)}}{2}, k = \overline{1, n-1}\}$ to find the possible minimum of $L(\hat{\mu}^a, \theta)$. Here $X_{(k)}$ is a k-th element in a sorted sample.

Decision tree router

Now we slightly increase router's complexity adding one more prameter:

$$p^2_{i:n}(\theta_1, \theta_2) = \begin{cases} 1, \ X_i > \theta_2 \\ 0.5, \ X_i \in [\theta_1, \theta_2] \\ 0, \ X_i < \theta_1 \end{cases} . \tag{9}$$

In this case for one dimensional features we still can select best parameters by brute-force. This router function $p^2_{i:n}$ is not differentable buy theta as well, as router from previous paragraph.

Logistic router, one feature

Now we consider differentiable router function

$$p^2_{i:n}(\theta_1, \theta_2) = \frac{1}{1 + e^{\theta_1 + \theta_2 X_i}}.$$

This means the eigenvalues of Γ_n are differentiable and we can use Newton-Rhaphson method to optimize loss functions. From the other hand, we still can use brute-force methods here like in previouse paragraphs.

Logistic router, d features

In this case $\tilde{X} = (1, X_1, ..., X_d)$ and $\theta \in \mathbb{R}^{d+1}$. So the router is

$$p^2_{i:n}(\theta) = \frac{1}{1 + e^{\langle \theta, \tilde{X}_i \rangle}}.$$

Here we can not use brute-force and we need to take into a count a differentiability of Γ_n and its eigenvalues. So to fit parameters θ we should count gradients of $Q(\mu, \theta)$, what is possible.

Questions for future

- Are there ways to build nonparametric routers using gradient boostings or decision trees?
- Is it possible to use EM algorithm in this model?
- Can we use a multivariate mixture of gaussians here for a router?
- Is it possible to use conditional minimization with constrans? Can we use Karush–Kuhn–Tu conditions?
- Differentiability for the loss function

0.1 Modeling for mean estimators

To check asymptotic propertis we run a series of simulations.

In each experiment for a fixed sample range we build $N = 1000$ sample from a gaussian mixture. For each sample we evaluate estiamates $(\hat{\mu}_k^a, \hat{\theta}^a)$ and $(\hat{\mu}_k^p, \hat{\theta}^p)$ for vectors (μ_k, θ). Then for any mean estimator fo μ and router's parameters θ we cound average and variance.

Component indices $\kappa_i = \kappa(O_i), i = 1, ..., n$ are taken from the distribution $(p_1, ..., p_M)$ where $\sum_{k=1}^{m} p_k = 1$.

Observed features $X_i^{(m)}$ from m-th mixture component had the next distribution:

$$X_i^{(m)} \simeq N(\mu_{(m)}, \sigma_{(m)}^2),$$

Distribution parameters are listed in table 0.1. The sample histogram is on the figure (0.1).

	Component	
	1	2
p_m	$\frac{1}{3}$	$\frac{2}{3}$
$\mu_{(m)}$	-3	2
$\sigma_{(m)}$	1	0.5

Table 0.1. Modeling parameters

The router function is chosen due to experiment.

Decision trunc router

From the chart (0.2) and (0.3) we see that point of minima are different for these loss functions and these two routers are different. Also we observe these functionals are similar near the point of local minimum.

Decision tree router

Kmeans router with grey zone

The last two shapes from (0.4) has only small differnece. Max val is 3. Impossible cases are white. Back point are points of minima.

White diagonal on figure (0.4) are regions near the parameters which maker the Gramm matrix singular.

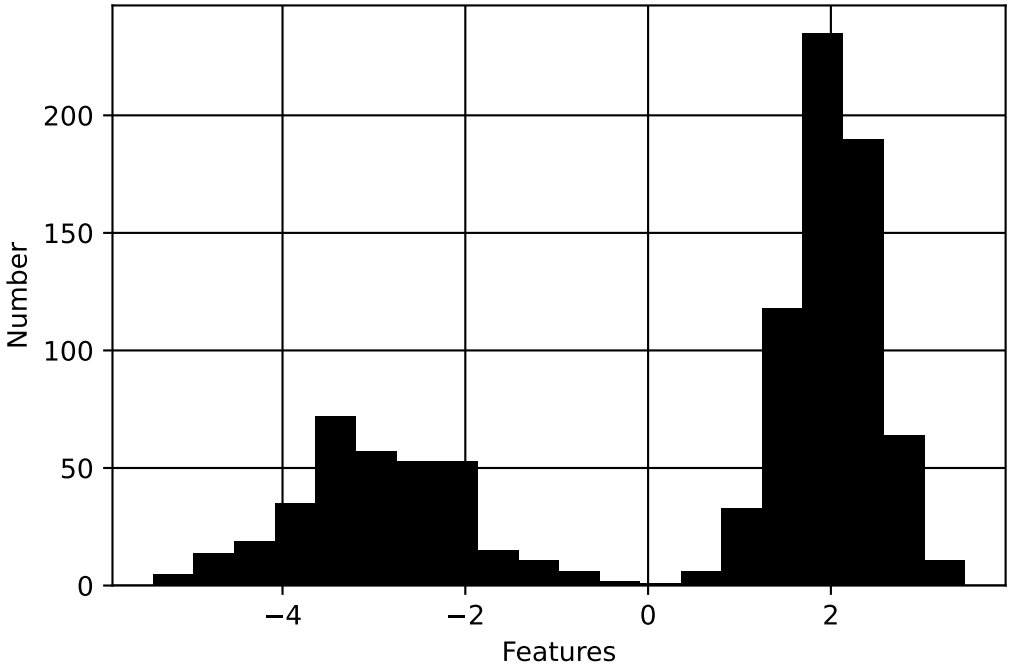

Figure 0.1. exmple of sample histogram for generated features from experiments

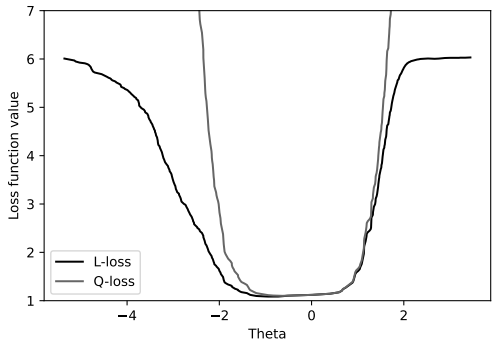

Figure 0.2. Decision trunc: $L(\theta, \hat{\mu}_k^a)$ and $Q(\theta, \hat{\mu}_k^p)$ loss functions

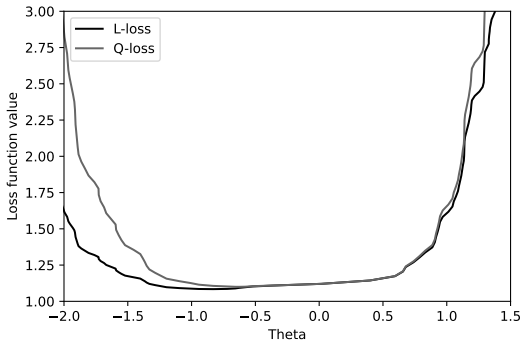

Figure 0.3. Decision trunc: $L(\theta, \hat{\mu}^a)$ and $Q(\theta, \hat{\mu}^p)$ loss functions

From this figure we observe different loss functions and different regions those containe point of local minimum.

Logistic router, 1D

Note: for this case we really do not need to use enequality (6) to build router. We can just brute-force parameter selection (the Decision trunc router works better then all other linear routers)

For this type of router we check the next equation (3). It was shown before for router

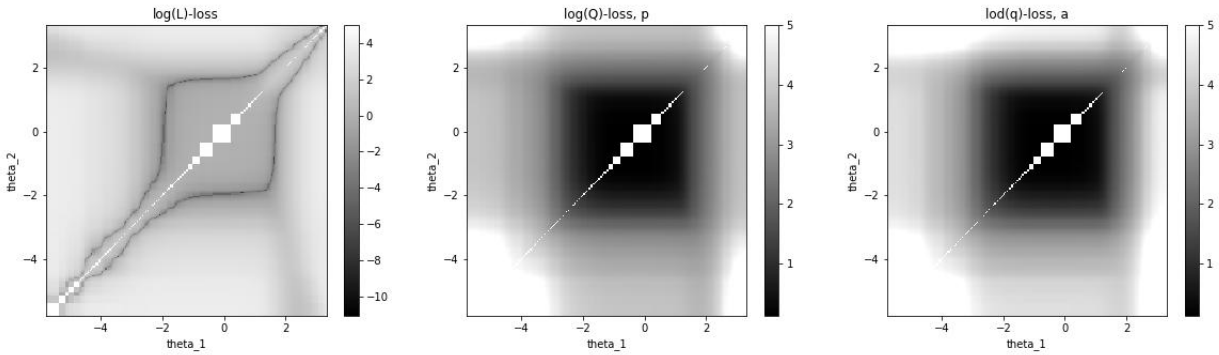

Figure 0.4. Decision tree: $\ln$ of $L(\theta, \hat{\mu}^a)$, $Q(\theta, \hat{\mu}^p)$ and $L(\theta, \hat{\mu}^a)$ loss functions

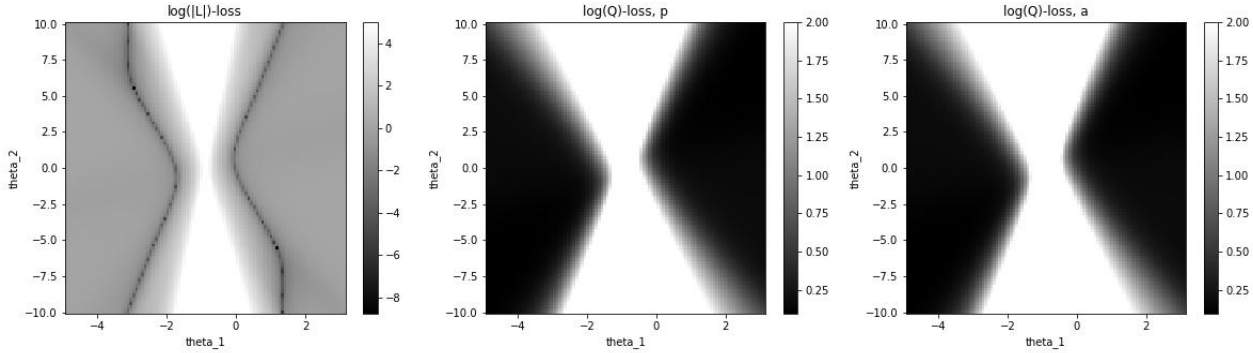

Figure 0.5. Logistic router: $\ln$ of $L(\theta, \hat{\mu}^a)$, $Q(\theta, \hat{\mu}^p)$ and $L(\theta, \hat{\mu}^a)$ loss functions

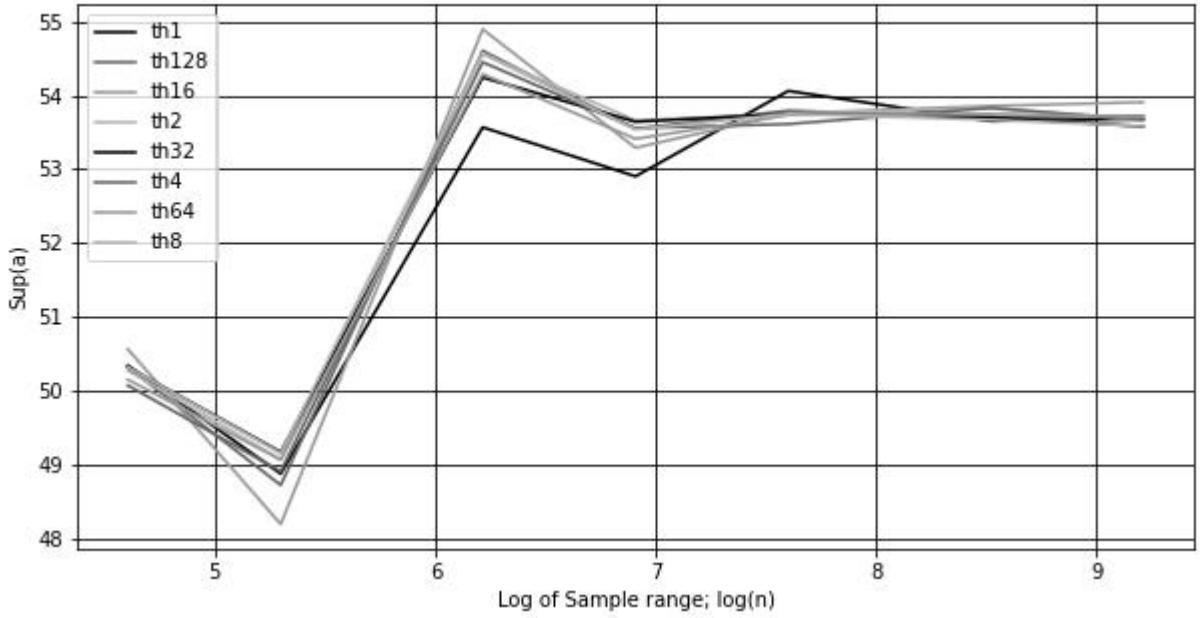

Figure 0.6. Logistic router: n vs $\sup |a| * n$ chart

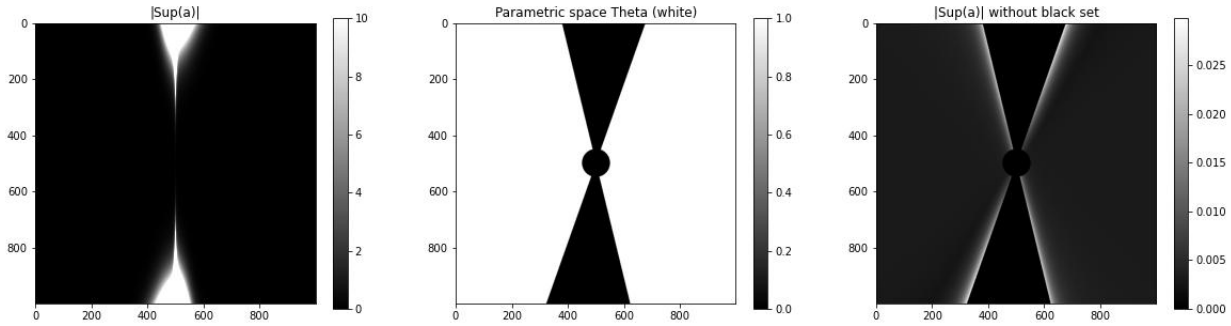

Figure 0.7. Logistic router: $\sup|a^k_{i:n}(\theta)|$ for $\theta \in \mathbb{R}^2$ (left) and parametric space Θ_r (center), $\sup|a^k_{i:n}(\theta)|$ chart

parameters except a ball near the zero-point the last equation holds. Also there are problems when only several objects belong to one class (as shown in counter-example in the beggining). Now we check this by series of simulating experiments.

we run series of $N = 1000$ experiments for different hyperparameters:

1 Sample size $n \in \{100, 200, 500, 1000, 2000, 5000, 10000\}$

2 With different ball radius $t \in \{1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128\}$. Weights near the zero-point has higher values. So the supremum should be higher

For this experimnets our paramteric space $\Theta_r \setminus \{B_t(0) \cup P_{0.05}\}$. In our simulation we throw away cases when less then 5% of sample belongs to one of the component (we call this set $P_{0.05}$). It is necessary ot use proportion because of the issue that was described in counter-exapmple for minimax weights $a^k_{i:n}$ (see the fig. 0.7).

From the fig (0.6) we see that radius of the ball does not have sufficient impact.

As a result, from the Figure 0.5 we observe too many points of minima for loss function Q and they covers regions with point of minima of function L.

Logistic router, 2D

This is the case where we really need differintiability of left part of the enequality (6) to build router. For multidimensional θ brute-force is too slow.

We build 1000 samples and fit parameters (μ, θ) minimizing loss $Q(\mu, \theta)$. The true values are in the table 0.2

	Component	
	1	2
p_m	$\frac{1}{3}$	$\frac{2}{3}$
$\mu_{(m)}$	(-0.5, -3)	(1, 3)
$\Sigma_{(m)}$	$\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$	$\begin{matrix} 1 & 0.5 \\ 0.5 & 1 \end{matrix}$

Table 0.2. Modeling parameters

Scatterplots 0.8 and 0.9 shows that there is no unique solutions. or this two cases the loss function are the almost the same. The histogram 0.10 shows that loss function is slightly dependent on initialization.

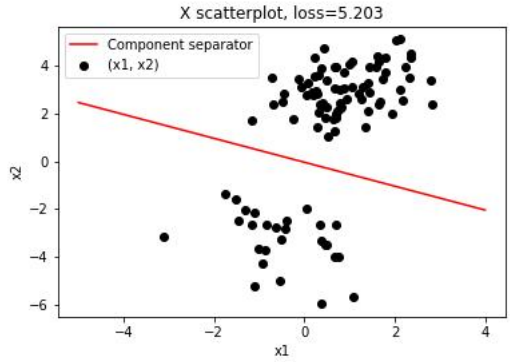

Figure 0.8. Feature scatterplot and separator. Case 1

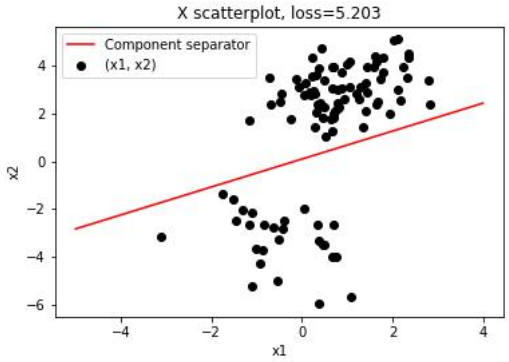

Figure 0.9. Feature scatterplot and separator. Case 2

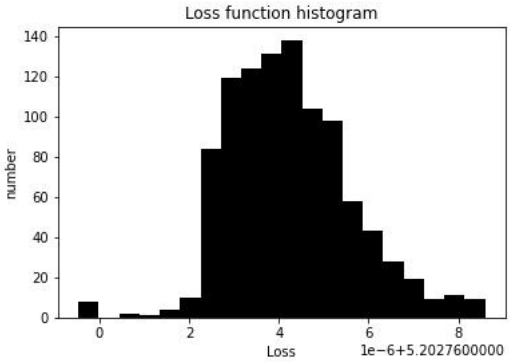

Figure 0.10. Histogram of the loss Q distribution for different starting points

So from the Figures 0.8 and 0.9 we observe that build router is capable to linearly separate clusters and this separation is not unique. The figure 0.10 shows small variance of the loss function Q for different randomly initialized parameters.

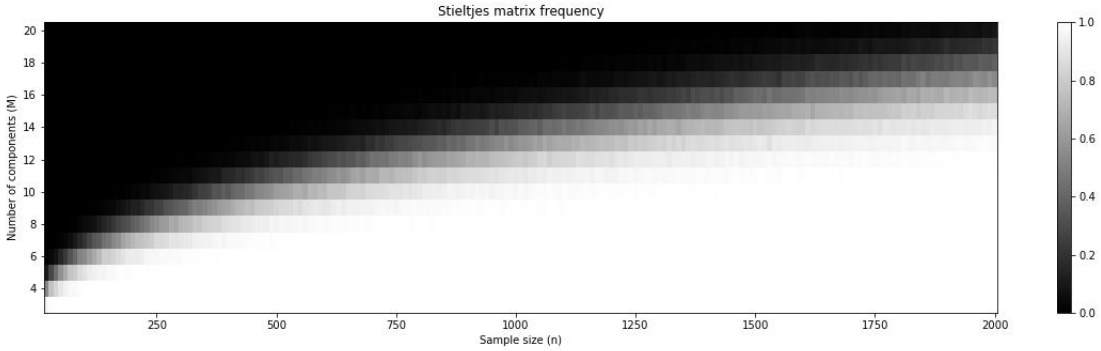

Figure 0.11. Frequency to obtain a Stieltjes matrix using mixing probabilities

Stieljes property check

At this section we chech the frequncy to obtain the Stieltjes matrix in different router schemas. This property is necessary to use enequality (8) for $M \geqslant 3$.

For any $n = 20, \bar{1}000$ and $M = 1, \bar{2}0$ we made 1000 experinemnts generating samples using different approaches.

Uniform sampling Here we build the matrix $P_n = (p^k_{i:n})^{k=1,\bar{M}}_{i=1,\bar{n}}$ using unifrom distrubutions and following schema.

1 Generate $u_{i,k} \simeq U[0,1]$ for $i = 1,\bar{n}$, $k = 1,\bar{M}$

2 Count $p^k_{i:n} = \frac{u_{i,k}}{\sum_{t=1}^{M} u_{i,t}}$ for $i = 1,\bar{n}$, $k = 1,\bar{M}$

Then we count $\Gamma_n^{-1} = (P_n P_n^T)^{-1}$ and count frequency of the maximum off-diagonal element to be positive. This is the case, when the Γ_n^{-1} is not a Stieltjes matrix.

The plot 0.11 shows that for a fixed M, the probability to obtain Stieltjes matrix goes to one with $n \to \infty$. From the other hand, we observe high probability to have non-Stieltjes matrix Γ_n^{-1} for large M values. For real applications the value M rarely is greater then 5 and samples usually contain more then 500 objects.

Conclusions

At this work we covered building mixture of experts using ideas of k-means clusterizer and minimax weighted measures. Alwe we proved several thorems for loss function boundaries and build more clusterizers using thes enequality. Through several sequence of experiment we have shown problems and posibilities of the new approach.

Bibliography

1. Pearson, K. (1894). Contributions to the mathematical theory of evolution. Phil. Trans. Roy. Soc. London A 185 , 71-110.
2. Hall P., Titterington D. M. The Use of Uncategorized Data to Improve the Performance of a Nonparametric Estimator of a Mixture Density // Journal of the Royal Statistical Society. Series B (Methodological), Vol. 47, No. 1 (1985), pp. 155-163
3. Dempster A. P., Laird N. M., Rubin D. B Maximum-likelihood from incomplete data via the em algorithm // J. Royal Statist. Soc. Ser. B., 39, 1977.
4. Jacobs, Robert A.; Jordan, Michael I.; Nowlan, Steven J.; Hinton, Geoffrey E. (February 1991). "Adaptive Mixtures of Local Experts". Neural Computation. 3 (1): 79–87.
5. Nowlan, Steven; Hinton, Geoffrey E (1990). "Evaluation of Adaptive Mixtures of Competing Experts". Advances in Neural Information Processing Systems. 3. Morgan-Kaufmann.
6. Maiboroda R.E., Sugakova O.V. Estimation and classification by the obseravations from a mixture // Kyiv University, (2009)
7. Liubashenko D., Maiboroda R. Linear regression by observatons from mixture with varying concentrations // Modern Stochastics: Theory and Applications, 2 ,No 4, 343 – 353, (2015)
8. Maiboroda R. E., Sugakova O. V. Statistics of mixtures with varying concentrations with application to DNA microarray data analysis // Journal of nonparametric statistics. 24 , No 1 201–205 (2012)
9. Maiboroda, R., Miroshnychenko, V., and Sugakova, O. Estimation of Concentrations Parameters in the Model of Mixture with Varying Concentrations. Austrian Journal of Statistics, 54(1), 1–16. Retrieved from https://ajs.or.at/index.php/ajs/article/view/1953
10. Abraham Berman, Robert J. Plemmons. Classics in applied mathematics, Society for Industrial and Applied Mathematics, 1987
11. Bilmes J. A. A gentle tutorial of the EM algorithm and its Application to Parameter

Estimation for Gaussian mixture and Hidden Markov Models // U.C Berkeley 1998

12. Maiboroda R. E., Sugakova O. V., Doronin A. V. Generalized estimating equations for mixtures with varying concentrations // The Canadian Journal of Statistics 41(2), 217–236 (2013)
13. Tzon-Tzer, Lu; Sheng-Hua, Shiou (2002). "Inverses of 2x2 block matrices". Computers & Mathematics with Applications. 43 (1–2): 119–129. doi:10.1016/S0898-1221(01)00278-4.
14. Lloyd, Stuart P. (1982), "Least squares quantization in PCM", IEEE Transactions on Information Theory, 28 (2): 129–137, doi:10.1109/TIT.1982.1056489, S2CID 10833328