統計的機械学習論
7. EMアルゴリズム

夏季集中講義(2019/09/02-13)

@九州工業大学

(飯塚キャンパス 大学院セミナー室７F)

このクラスのねらいと達成目標

ねらい

教師なしクラスタリングのアルゴリズムとして最尤推定法を利用したEMアルゴリズムを理解する。

達成目標

- **最尤推定法がEMアルゴリズムの中でどのように適用されているかを知る**
- **画像生成器とは何か、画像生成器が果たす役割について理解する**

クラスの進行

7. EMアルゴリズム：最尤推定法による教師なし学習

7.1. ベルヌーイ分布を用いた最尤推定法

7.1.1.手書き文字の合成方法

7.2.混合分布を用いた最尤推定法

7.2.1.混合分布による確率の計算

7.2.2.EMアルゴリズムの手続き

7.2.3.サンプルコードによる確認

7.2.4.クラスタリングによる探索的なデータ解析

7.3.手書き文字データの入手方法

7 EMアルゴリズム

最尤推定法による教師なし学習

手書き文字の分類を例にEMアルゴリズムを紹介する。

EMアルゴリズムを理解するための２つのステップ

- **「代表文字の生成」**

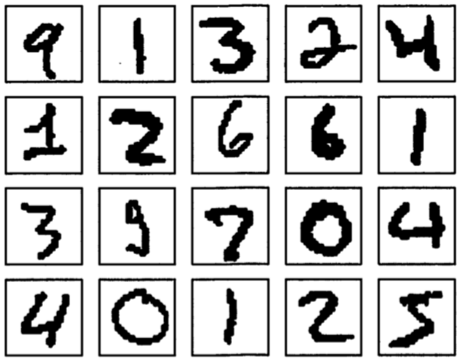
  ベルヌーイ分布を用いた最尤推定法の実施

- **「文字サンプルの分類」**

  混合ベルヌーイ分布を用いた最尤推定法を使う

7.1 ベルヌーイ分布を用いた最尤推定法

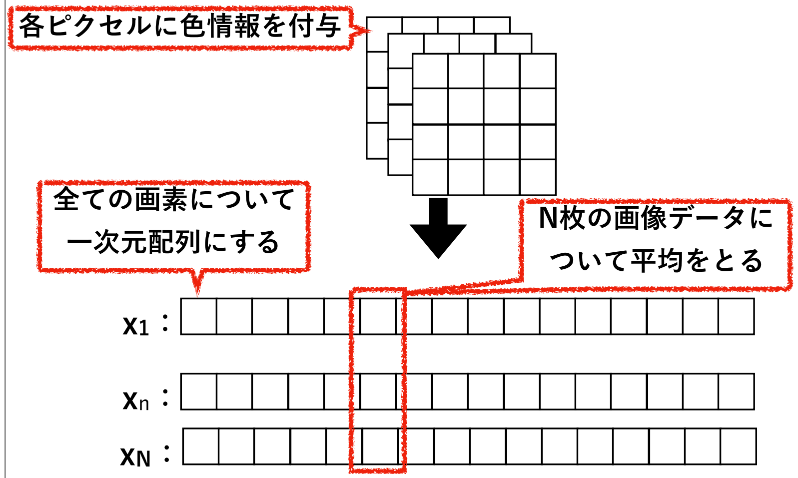
手書き文字の画像データを使おう



複数の画像を合成して代表文字（平均的な文字）を生成する

7.1.1 手書き文字の合成方法

画像データの扱い方



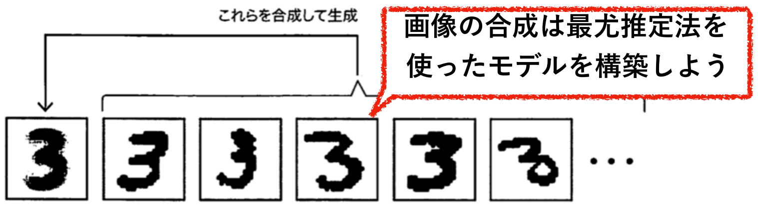
7.1.1 手書き文字の合成方法

$\boldsymbol{\mu}$：画像ベクトル（n=1, 2, …, N）の平均をとったベクトル

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$\boldsymbol{\mu}$の各成分はピクセルの色の濃淡を示す数値

（0から1の範囲で0なら「白」1なら「黒」というように）



7.1.2 「画像生成器」による最尤推定法の適用

最尤推定法はトレーニングセットが得られる確率を
最大にするパラメータを求める手法

⬇

「トレーニングセットを得る確率」が必要

⬇

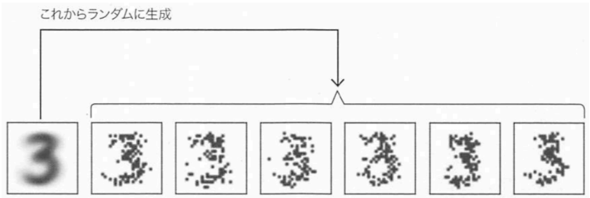
画像生成器を用意してランダムに手書き文字を生成し
トレーニングセットの手書き文字と一致する確率を求めよう

7.1.2 「画像生成器」による最尤推定法の適用

ベクトル$\boldsymbol{\mu}$の各成分μ_iの値＝対応するピクセルが黒になる確率

各要素に0~1の範囲で値を付与
0なら「白」1なら「黒」

$\boldsymbol{\mu}$:



トレーニングセット：N枚の手書き画像データ

画像生成器からN枚の手書き画像を生成しトレーニングセットの手書き文字と一致する確率が
最大となる画像生成器を見つけ出す

7.1.2 「画像生成器」による最尤推定法の適用

トレーニングセットのデータ群が生成される確率を求めよう

$\mathbf{x}$ **：特定の画像データ**

x_i **：$\mathbf{x}$のi番目のピクセルの色**

D **：画像のピクセル数**

P_i **：i番目のピクセルの色が得られる確率**

$$\begin{cases} x_i = 1 \to P_i = \mu_i \\ x_i = 0 \to P_i = 1 - \mu_i \end{cases}$$

7.1.2 「画像生成器」による最尤推定法の適用

確率をまとめて記述

$$P_i = \mu_i^{x_i}(1-\mu_i)^{1-x_i}$$

全てのピクセルについて考えると（ピクセル数D）

$$P(\mathbf{x}) = \prod_{i=1}^{D} P_i = \prod_{i=1}^{D} \mu_i^{x_i}(1-\mu_i)^{1-x_i}$$

全てのピクセルで同じ色になる確率

全てのトレーニングセット$\{x_n\}_{n=1}^{N}$について考えると

$$\boxed{P(\mathbf{x}) = \prod_{n=1}^{N} P(\mathbf{x}_n) = \prod_{n=1}^{N}\prod_{i=1}^{D} \mu_i^{[\mathbf{x}_n]_i}(1-\mu_i)^{1-[\mathbf{x}_n]_i}}$$

尤度関数

7.1.2 「画像生成器」による最尤推定法の適用

尤度関数を最大にする画像生成器を以下のように考えるが、

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n$$

ここで改めて尤度関数を最大化する$\boldsymbol{\mu}$について考える

$$\ln P = \sum_{n=1}^{N}\sum_{i=1}^{D}\{[\mathbf{x}_n]_i \ln \mu_i + (1-[\mathbf{x}_n]_i)\ln(1-\mu_i)\}$$

μ_iで偏微分すると、

$$\frac{\partial \ln P}{\partial \mu_i} = \sum_{n=1}^{N}\left(\frac{[\mathbf{x}_n]_i}{\mu_i} - \frac{1-[\mathbf{x}_n]_i}{1-\mu_i}\right)$$

7.1.2 「画像生成器」による最尤推定法の適用

$$\frac{\partial \ln P}{\partial \mu_i} = \sum_{n=1}^{N} \left(\frac{[\mathbf{x}_n]_i}{\mu_i} - \frac{1-[\mathbf{x}_n]_i}{1-\mu_i} \right)$$

偏微分係数が0になる場所が極大値となるため、
上記式=0について解けば

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n$$

7.1.2 「画像生成器」による最尤推定法の適用

ところで、確率を導く式がロジスティック回帰でのt_nのデータが得られる式と似ている

$$P(\mathbf{x}) = \prod_{i=1}^{D} P_i = \prod_{i=1}^{D} \mu_i^{x_i}(1-\mu_i)^{1-x_i}$$

$$\Updownarrow$$

$$P = \prod_{n=1}^{N} P_n = \prod_{n=1}^{N} P_n^{t_n}(1-P_n)^{1-t_n}$$

このような確率分布を**ベルヌーイ確率分布**という

7.2 混合分布を用いた最尤推定法

複数の数字の手書き画像データが混在したデータセットから文字の種類ごとに分類する方法

7.2.1 混合分布による確率の計算

K：データセットに含まれる数字の種類

$\{\boldsymbol{\mu}_k\}_{k=1}^{K}$：各種類ごとに用意する画像生成器

$$P_{\mu_k}(\mathbf{x}) = \prod_{i=1}^{D} [\boldsymbol{\mu}_k]_i^{x_i}(1-[\boldsymbol{\mu}_k]_i^{x_i})^{1-x_i}$$

特定の画像生成器から画像と同じ画像が生成される確率

7.2.1 混合分布による確率の計算

ここでK個の画像生成器の中からランダムに1つの画像生成器を選択すると考える。

k番目の画像生成器を選択する確率：

$$\{\pi_k\}_{k=1}^{K} \qquad \sum_{k=1}^{K}\{\pi_k\}_{k=1}^{K} = 1$$

特定の画像$\mathbf{x}$と同じ画像が生成される確率は
=k番目の画像生成器を選択し、さらに画像$\mathbf{x}$を生成する確率

$$p(\mathbf{x}) = \sum_{k=1}^{K}\{\pi_k\} p_{\boldsymbol{\mu}_k}(\mathbf{x})$$

7.2.1 混合分布による確率の計算

トレーニングセットにはN枚の画像データがあるなら、

$$P=\prod_{n=1}^{N}p(\mathbf{x}_n)=\prod_{n=1}^{N}\sum_{k=1}^{K}\{\pi_k\}p_{\boldsymbol{\mu}_k}(\mathbf{x}_n)$$

尤度関数

画像生成器の持つ確率分布は

ベルヌーイ確率分布に従う

k個の画像生成器が持つ、それぞれのベルヌーイ分布が混在するため、「**混合ベルヌーイ分布**」と呼ばれる。

7.2.2 EMアルゴリズムの手続き

今回は対数にしても総乗と総和が混在しており、計算が複雑。

$$P(\mathbf{x})=\prod_{n=1}^{N}P(\mathbf{x}_n)=\prod_{n=1}^{N}\prod_{i=1}^{D}\mu_i^{[\mathbf{x}_n]_i}(1-\mu_i)^{1-[\mathbf{x}_n]_i}$$

尤度関数

$$P=\prod_{n=1}^{N}p(\mathbf{x}_n)=\prod_{n=1}^{N}\sum_{k=1}^{K}\{\pi_k\}p_{\boldsymbol{\mu}_k}(\mathbf{x}_n)$$

EMアルゴリズムはこのような形の尤度関数を最大にするパラメータを求める手続きである。

7.2.2 EMアルゴリズムの手続き

EMアルゴリズムは少しk平均法と似ているので、類似点を思い出しながら解説する。

最初にk個の画像生成器$\{\boldsymbol{\mu}_k\}_{k=1}^{K}$を適当に設定する。
（kmeansでいうところの代表点を決める過程）

$$p(\mathbf{x}_n)=\sum_{k=1}^{K}\pi_k p_{\boldsymbol{\mu}_k}(\mathbf{x}_n)$$

k番目の画像生成器を選択する確率

$\mathbf{x}_n$と同じ画像を生成する確率

どの画像生成器を使っても$\mathbf{x}_n$と同じ画像が生成される可能性がある

7.2.2 EMアルゴリズムの手続き

k番目の画像生成器から$\mathbf{x}_n$と同じ画像を生成する確率を求める

$$\gamma_{nk}=\frac{\pi_k p_{\boldsymbol{\mu}_k}(\mathbf{x}_n)}{\sum_{k'=1}^{K}\pi_{k'}p_{\boldsymbol{\mu}_{k'}}(\mathbf{x}_n)}$$

（k平均法の各データが所属する代表点を決める過程）

今回、$\mathbf{x}_n$はどれか１つの画像生成器にのみ所属しているのではなく、γ_{nk}の割合で所属していると考える。
γ_{nk}が決定したら新しい画像生成器を作り直す。

$$\boldsymbol{\mu}_k=\frac{\sum_{n=1}^{N}\gamma_{nk}\mathbf{x}_n}{\sum_{n=1}^{N}\gamma_{nk}}\qquad \pi_k=\frac{\sum_{n=1}^{N}\gamma_{nk}}{N}$$

7.2.3 サンプルコードによる確認

「07-prep_data」を使ってデータを準備する。

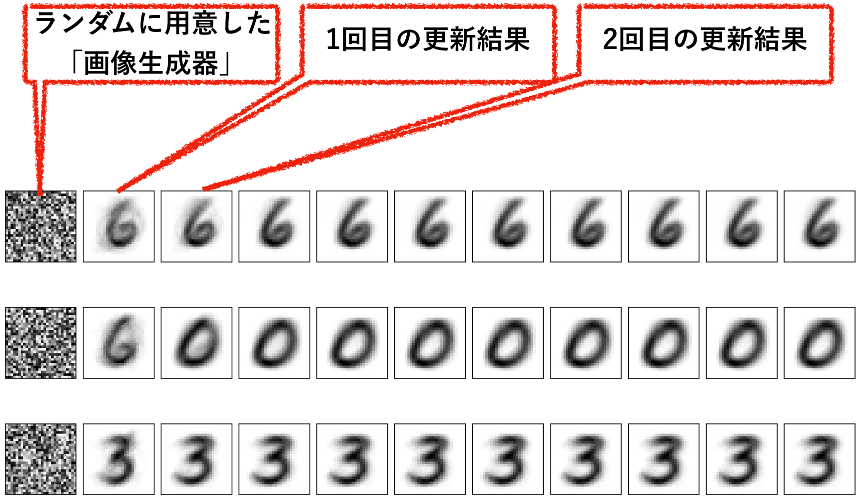
生成されるデータ：

「sample-images.txt」「sample-labels.txt」

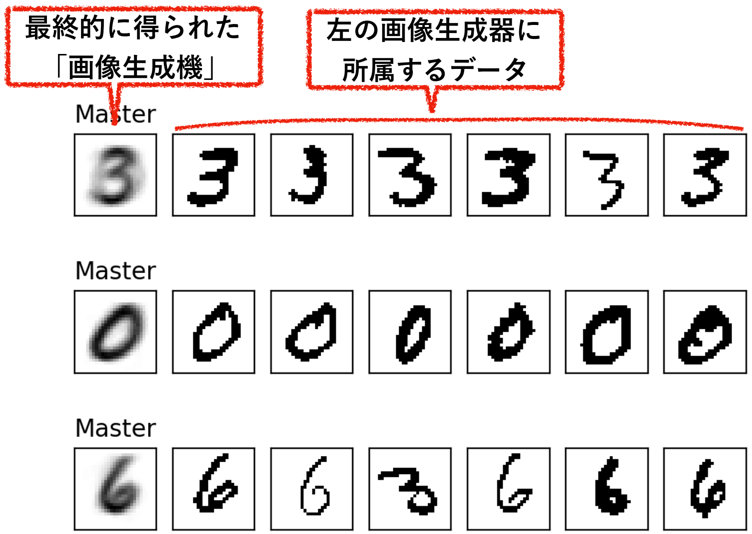
参考として、最初の１０文字について形状をアスキーアートで表現したものが「samples.txt」で確認できる。

7.2.3 サンプルコードによる確認

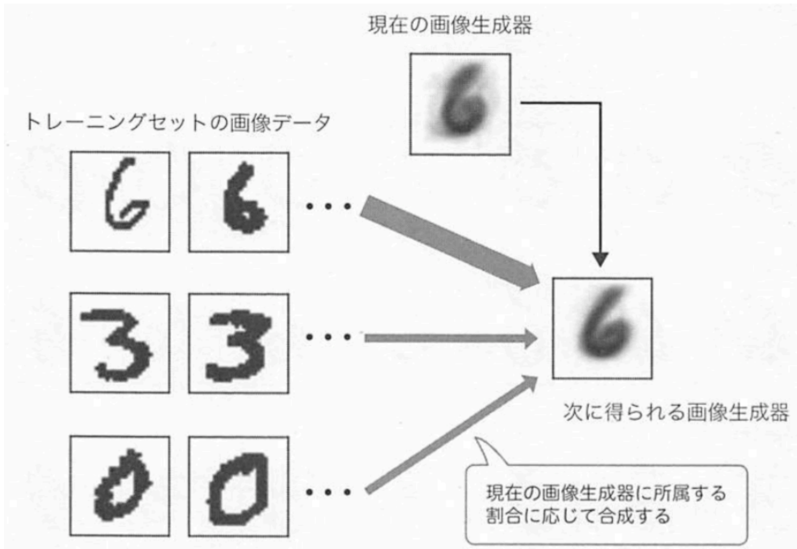
「07-mix_em」を使ってデータを準備する。



7.2.3 サンプルコードによる確認



7.2.3 サンプルコードによる確認



まとめ

- EMアルゴリズムにおける画像生成器の役割を理解できた
- k平均法を下地にしたEMアルゴリズムの手続きを知った
- EMアルゴリズムにおける尤度関数、対数尤度関数を記述できる

ミニッツペーパー(9/9分)

- 画像生成器とは何か、EMアルゴリズムにおける役割を説明してください。

  画像生成器は画像データの各ピクセルの色情報を確率分布という形で表現したものである。
  EMアルゴリズムではトレーニングセットの画像データがどの画像生成器にどのくらいの割合で所属するかを計算し、所属する割合の高い画像データに重み付けを行う。そしてトレーニングセットの画像と同じ画像となる確率が最大になるように画像生成器を更新する。