Московский государственный университет имени М. В. Ломоносова

Байсовские методы машинного обучения

Задание 2. EM алгоритм для детектива

Выполнил:
студент 4 курса 417 группы
Бабичев Дмитрий Олегович

Москва, 2017

Содержание

1 Описание модели

Дана выборка $\boldsymbol{X} = \{\boldsymbol{X_k}\}_{k=1}^K$ сильно зашумленных черно-белых изображений размера $H \times W$ пикселей. Каждое из этих изображений содержит один и тот же неподвижный фон и лицо преступника в неизвестных координатах, при этом лицо попадает в любое изображение целиком. Будем считать, что изображение лица имеет прямоугольную форму размера $h \times w$ пикселей. Значения $h,\ w$ в выданных данных указаны в описании задания в anytask.

Введем следующие обозначения:

- $\boldsymbol{X_k}(i,j)$ – пиксель k-ого изображения;
- $\boldsymbol{B} \in \mathbb{R}^{H \times W}$ – изображение чистого фона без лица преступника, $\boldsymbol{B}(i,j)$ – пиксель этого изображения;
- $\boldsymbol{F} \in \mathbb{R}^{h \times w}$ – изображение лица преступника, $\boldsymbol{F}(i,j)$ – пиксель этого изображения;
- $\boldsymbol{d}_k = (d_k^h, d_k^w)$ – координаты верхнего левого угла изображения лица на k-ом изображении (d_k^h – по вертикали, d_k^w – по горизонтали), $\boldsymbol{d} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K)$ – набор координат для всех изображений выборки.

Также будем считать шум на изображении независимым для каждого пикселя и принадлежащим нормальному распределению $\mathcal{N}(0, s^2)$, где s – стандартное отклонение. Таким образом для одного изображения имеем:

$$p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) = \prod_{ij} \begin{cases} \mathcal{N}(\boldsymbol{X_k}(i,j) \mid \boldsymbol{F}(i - d_k^h, j - d_k^w), s^2), & \text{если } (i,j) \in faceArea(\boldsymbol{d}_k) \\ \mathcal{N}(\boldsymbol{X_k}(i,j) \mid \boldsymbol{B}(i,j), s^2), & \text{иначе} \end{cases},$$

где $\boldsymbol{\theta} = \{\boldsymbol{B}, \boldsymbol{F}, s^2\}$, $faceArea(\boldsymbol{d}_k) = \{(i,j) \mid d_k^h \le i \le d_k^h + h - 1,\ d_k^w \le j \le d_k^w + w - 1\}$.

Распределение на неизвестные координаты лица на изображении зададим общим для всех изображений с помощью матрицы параметров $\boldsymbol{A} \in \mathbb{R}^{(H-h+1)\times(W-w+1)}$ следующим образом:

$$p(\boldsymbol{d}_k \mid \boldsymbol{A}) = \boldsymbol{A}(d_k^h, d_k^w), \qquad \sum_{ij} \boldsymbol{A}(i,j) = 1,$$

где $\boldsymbol{A}(i,j) = a_{ij}$ – элемент матрицы $\boldsymbol{A}$.

В итоге имеем следующую совместную вероятностную модель:

$$p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) = \prod_k p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) p(\boldsymbol{d}_k \mid \boldsymbol{A}).$$

2 Необходимые формулы

Попробуем расписать подробно классический EM-алгоритм для этой задачи. В текущей формулировке скрытыми переменными являются координаты левого верхнего угла изображения с лицом. На E-шаге нужно вычислить их апостериорную оценку $q(\boldsymbol{d})$:

$$q(\boldsymbol{d}) = p(\boldsymbol{d} \mid \boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{A}) = \prod_k p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A}).$$

Воспользовавшись формулой условной вероятности $p(A|B) = \dfrac{p(A,B)}{p(B)}$ и правилом суммирования $p(B) = \int\limits_A p(A,B)dA$, можно получить следующее выражение $p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})$:

$$p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A}) = \frac{p(\boldsymbol{X}_k, \boldsymbol{d}_k \mid \boldsymbol{\theta}, \boldsymbol{A})}{p(\boldsymbol{X}_k \mid \boldsymbol{\theta}, \boldsymbol{A})} = \frac{p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) p(\boldsymbol{d}_k \mid \boldsymbol{A})}{\sum_{i=0}^{H-h} \sum_{j=0}^{W-w} p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij} = (i,j), \boldsymbol{\theta}) p(\boldsymbol{d}_{ij} = (i,j) \mid \boldsymbol{A})}.$$

При фиксированных параметрах $\boldsymbol{\theta}, \boldsymbol{A}$ все необходимые для вычислений величины нам известны.

На M-шаге нам нужно вычислить точечные оценки $\boldsymbol{\theta}, \boldsymbol{A}$:

$$\mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) \ \rightarrow \ \max_{\boldsymbol{\theta}, \boldsymbol{A}}, \qquad \sum_{i=1}^{H} \sum_{j=1}^{W} a_{ij} = 1.$$

Распишем подробнее:

$$\begin{aligned}
&\mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) = \mathbb{E}_{q(\boldsymbol{d})} \sum_{k=1}^{K} \log p(\boldsymbol{X}_k, \boldsymbol{d}_k \mid \boldsymbol{\theta}, \boldsymbol{A}) = \\
&= \sum_{k=1}^{K} \mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}_k, \boldsymbol{d}_k \mid \boldsymbol{\theta}, \boldsymbol{A}) = \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q^k(\boldsymbol{d}_{ij}) \log p(\boldsymbol{X}_k, \boldsymbol{d}_{ij} \mid \boldsymbol{\theta}, \boldsymbol{A}) = \\
&\qquad\qquad = \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k (\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij}, \boldsymbol{\theta}) + \log a_{ij})
\end{aligned}$$

Очевидно, что пытаться максимизировать по a_{ij} бесполезно, ведь чем эти коэффициенты больше, тем лучше. Но к счастью, у нас задача с ограничениями в виде равенства. Запишем для нее Лагранжиан:

$$L(\boldsymbol{\theta}, \boldsymbol{A}, \lambda) = \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k (\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij}, \boldsymbol{\theta}) + \log a_{ij}) - \lambda (\sum_{i=1}^{H} \sum_{j=1}^{W} a_{ij} - 1).$$

Перейдем к системе:

$$\begin{cases}
\dfrac{\partial L}{\partial a_{ij}} = \dfrac{\sum_{k=1}^{K} q_{ij}^k}{a_{ij}} - \lambda = 0 \quad \Rightarrow \quad a_{ij} = \dfrac{\sum_{k=1}^{K} q_{ij}^k}{\lambda}, \\
\dfrac{\partial L}{\partial \boldsymbol{\theta}} = 0 \qquad\qquad \text{распишем позже}, \\
\dfrac{\partial L}{\partial \lambda} = \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} a_{ij} - 1 = 0 \quad \Rightarrow \quad \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} a_{ij} = 1,
\end{cases} \Longrightarrow \lambda = \sum_{k=1}^{K} \sum_{i=1}^{H} \sum_{j=1}^{W} q_{ij}^k \Rightarrow a_{ij} = \frac{\sum_{k=1}^{K} q_{ij}^k}{K}.$$

Вспомним, что $\boldsymbol{\theta} = \{\boldsymbol{B}, \boldsymbol{F}, s^2\}$. Трудностей с дифференцированием суммы логарифмов двух нормальных распределений нет, лично у меня возникла проблема с адекватной записью пределов суммирования. Если кратко, то b_{ij} и f_{ij} представляют собой усреднения по всем изображениям и по всевозможным положениям лица на изображениях соответствующих элементов $\boldsymbol{X}_k(i,j)$ с весами q_{ij}^k. Запишем наш Лагранжиан без независящих от $\boldsymbol{\theta}$ членов (BG - background, FA - face area):

$$L(\boldsymbol{B}, \boldsymbol{F}, s) = \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k \log p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij}, \boldsymbol{\theta}) =$$

$$= \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k \Bigg[\Bigg(\sum_{\substack{l,m \in \\ FA(i,j)}} -\log s - \frac{1}{2} 2\pi - \frac{1}{2s^2} (x_{l,m}^k - f_{l-i,m-j})^2 \Bigg) +$$

$$+ \Bigg(\sum_{\substack{l,m \in \\ BG(i,j)}} -\log s - \frac{1}{2} 2\pi - \frac{1}{2s^2} (x_{l,m}^k - b_{l,m})^2 \Bigg) \Bigg].$$

Распишем производные по элементам матриц $\boldsymbol{B}, \boldsymbol{F}$ и по s:

$$\begin{cases} \dfrac{\partial L}{\partial b_{l,m}} = \sum\limits_{k=1}^{K} \sum\limits_{i=1}^{H-h} \sum\limits_{j=1}^{W-w} \dfrac{q_{ij}^k}{s^2} (x_{l,m}^k - b_{l,m}) I\left[l, m \in BG(i,j)\right] = 0, \\ \dfrac{\partial L}{\partial f_{l,m}} = \sum\limits_{k=1}^{K} \sum\limits_{i=1}^{H-h} \sum\limits_{j=1}^{W-w} \dfrac{q_{ij}^k}{s^2} (x_{i+l-1,j+m-1}^k - f_{l,m}) I\left[l, m \in FA(i,j)\right] = 0, \\ \dfrac{\partial L}{\partial s} = \sum\limits_{k=1}^{K} \sum\limits_{i=1}^{H-h} \sum\limits_{j=1}^{W-w} \dfrac{q_{ij}^k}{s} \left(\dfrac{1}{s^2} \left(\sum\limits_{\substack{l,m \in \\ FA(i,j)}} (x_{l,m}^k - f_{l-i,m-j})^2 + \sum\limits_{\substack{l,m \in \\ BG(i,j)}} (x_{l,m}^k - b_{l,m})^2 \right) - HW \right) = 0 \end{cases} \Longrightarrow$$

$$\begin{cases} b_{l,m} = \dfrac{\sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k x_{l,m}^k I\left[l, m \in BG(i,j)\right]}{\sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k I\left[l, m \in BG(i,j)\right]} \\ f_{l,m} = \dfrac{\sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k x_{i+l-1,j+m-1}^k I\left[l, m \in FA(i,j)\right]}{\sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k I\left[l, m \in FA(i,j)\right]} \\ s^2 = \dfrac{\sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k \left(\sum_{\substack{l,m \in \\ FA(i,j)}} (x_{l,m}^k - f_{l-i,m-j})^2 + \sum_{\substack{l,m \in \\ BG(i,j)}} (x_{l,m}^k - b_{l,m})^2 \right)}{KHW} \end{cases}$$

Осталось вывести лишь формулу нижней оценки $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A})$:

$$\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A}) = \mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) - \mathbb{E}_{q(\boldsymbol{d})} \log q(\boldsymbol{d}) =$$

$$= \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k (\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij}, \boldsymbol{\theta}) + \log a_{ij}) - \sum_{k=1}^{K} \sum_{i=1}^{H-h} \sum_{j=1}^{W-w} q_{ij}^k \log q_{ij}^k,$$

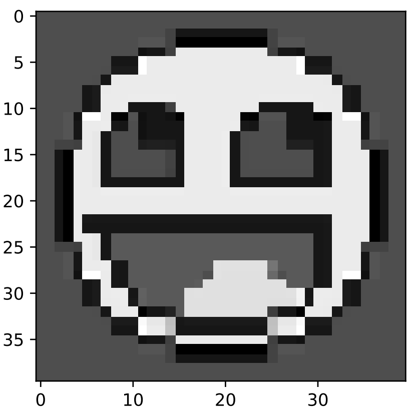
где

$$\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_{ij}, \boldsymbol{\theta}) = \Bigg[\Bigg(\sum_{\substack{l,m \in \\ FA(i,j)}} -\log s - \frac{1}{2} 2\pi - \frac{1}{2s^2} (x_{l,m}^k - f_{l-i,m-j})^2 \Bigg) + \Bigg(\sum_{\substack{l,m \in \\ BG(i,j)}} -\log s - \frac{1}{2} 2\pi - \frac{1}{2s^2} (x_{l,m}^k - b_{l,m})^2 \Bigg) \Bigg].$$

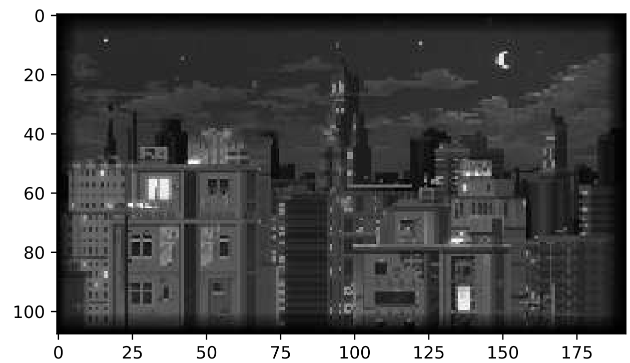
Зная все необходимые распределения вычисление этой оценки не составит труда.

Чтобы получить так называемый hard EM, необходимо преобразовать $q(\boldsymbol{d})$ так, что для каждого изображения $\boldsymbol{X_k}$ оценка $q(\boldsymbol{d}_k)$ принимает значений 1 только в точке максимума апостериорного распределения $p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})$. Можно конечно заного все расписать, введя новое обозначение для этого максимума, но на деле же в общем виде формулы не поменяются, добавится лишь один шаг между E и M - нахождение максимума и приведение каждого $q(\boldsymbol{d}_k)$ к соответствующему виду.

3 Краткий анализ



face



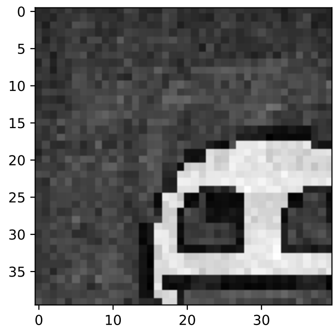
background

Пункт 1

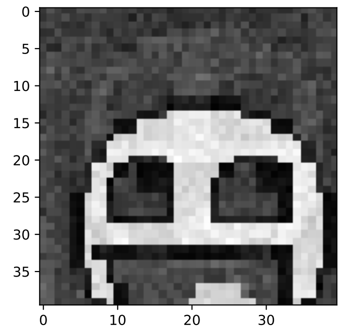
Протестируйте полученный EM алгоритм на сгенерированных данных. Сильно ли влияет начальное приближение на параметры на результаты работы? Стоит ли для данной задачи запускать EM алгоритм из разных начальных приближений?

В текущей реализации для каждого запуска перед первым E-шагом случайным образом генерируются величины $\boldsymbol{F}$, $\boldsymbol{B}$, $\boldsymbol{A}$, s. Посмотрим на результаты работы алгоритма после 12 запусков ($s_{real} = 30$):

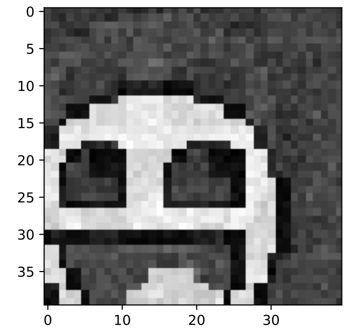
Очень странный результат. После каждого запуска смайлик вполне можно опознать, но вот его расположение меняется. То есть говорить о зависимости результата от начального приближения можно.



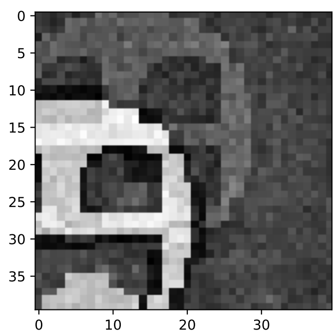
restart 1



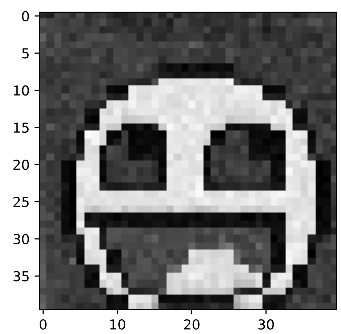
restart 2



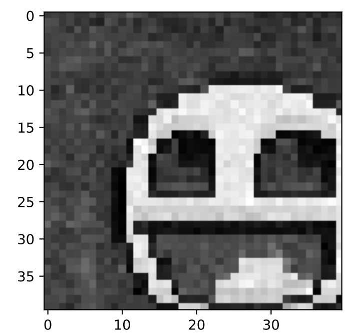
restart 3



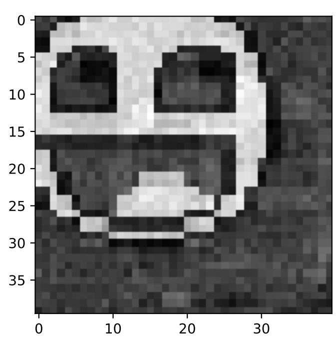
restart 4



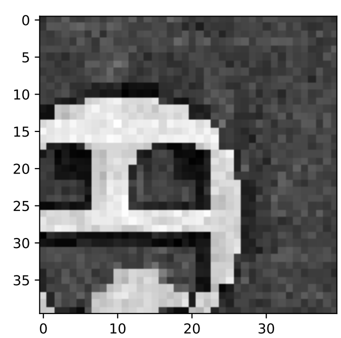
restart 5



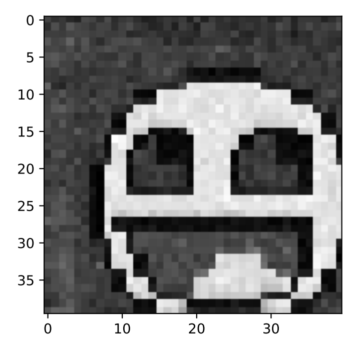
restart 6



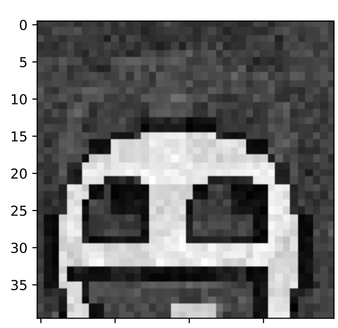
restart 7



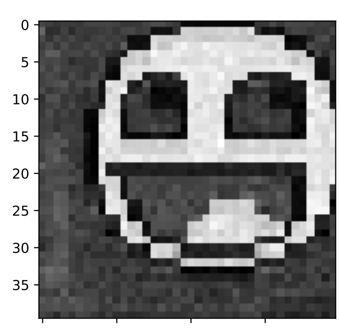
restart 8



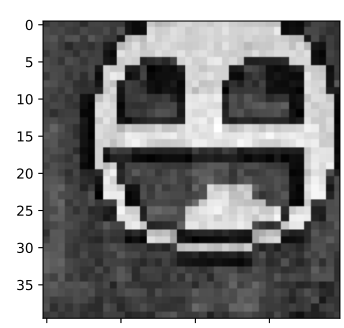
restart 9



restart 10



restart 11

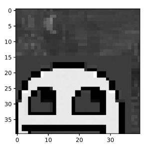
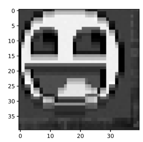
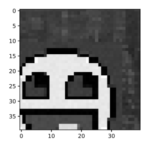
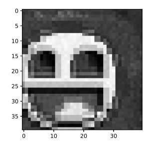
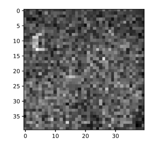
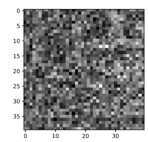
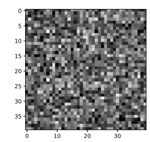
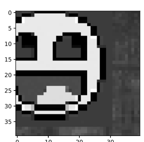
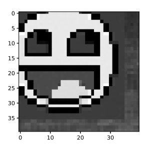
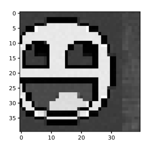
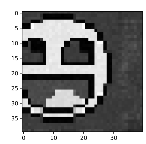
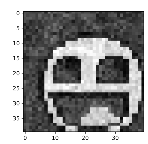
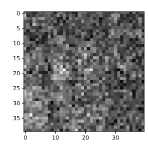
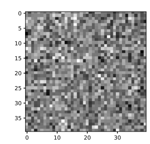
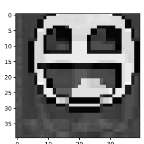
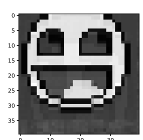
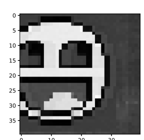
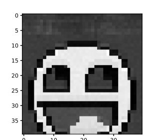
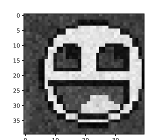
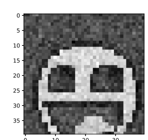
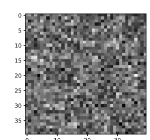
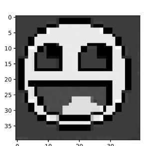
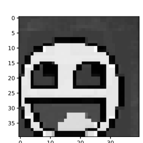
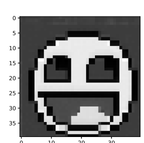
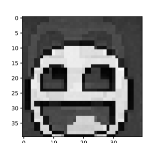
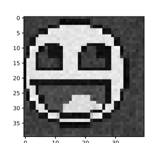
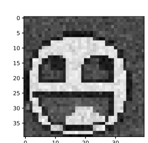
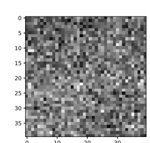
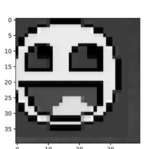
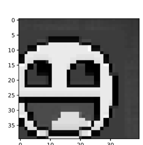
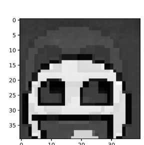
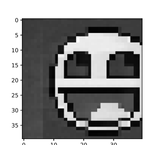
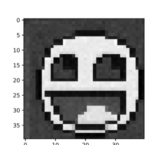
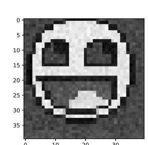
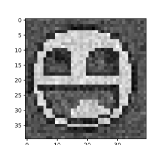


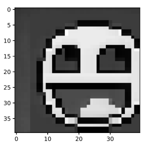
restart 12

Пункт 2

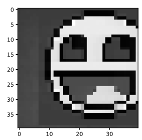
Запустите EM алгоритм на сгенерированных выборках разных размеров и с разным уровнем зашумления. Как изменения в обучающей выборке влияют на результаты работы (получаемые $\boldsymbol{F}$, $\boldsymbol{B}$ и $L(q, \boldsymbol{\theta}, \boldsymbol{A})$)? При каком уровне шума EM-алгоритм перестает выдавать вменяемые результаты? В данном пункте учтите, что для сравнения значения $L(q, \boldsymbol{\theta}, \boldsymbol{A})$ для выборок разного размера стоит нормировать его на объем выборки.

Рузельтат работы алгоритма для разных размеров выборки и разных дисперсий шума:

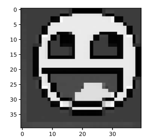
						
$K = 5$, $s = 1$	$K = 5$, $s = 2$	$K = 5$, $s = 5$	$K = 5$, $s = 10$	$K = 5$, $s = 50$	$K = 5$, $s = 100$	$K = 5$, $s = 200$
						
$K = 10$, $s = 1$	$K = 10$, $s = 2$	$K = 10$, $s = 5$	$K = 10$, $s = 10$	$K = 10$, $s = 50$	$K = 10$, $s = 100$	$K = 10$, $s = 200$
						
$K = 20$, $s = 1$	$K = 20$, $s = 2$	$K = 20$, $s = 5$	$K = 20$, $s = 10$	$K = 20$, $s = 50$	$K = 20$, $s = 100$	$K = 20$, $s = 200$
						
$K = 50$, $s = 1$	$K = 50$, $s = 2$	$K = 50$, $s = 5$	$K = 50$, $s = 10$	$K = 50$, $s = 50$	$K = 50$, $s = 100$	$K = 50$, $s = 200$
						
$K = 100$, $s = 1$	$K = 100$, $s = 2$	$K = 100$, $s = 5$	$K = 100$, $s = 10$	$K = 100$, $s = 50$	$K = 100$, $s = 100$	$K = 100$, $s = 200$



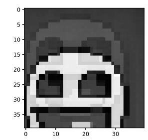
$K = 500,$ $s = 1$



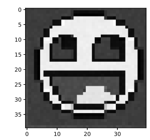
$K = 500,$ $s = 2$



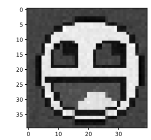
$K = 500,$ $s = 5$



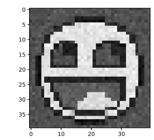
$K = 500,$ $s = 10$



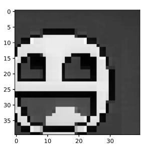
$K = 500,$ $s = 50$



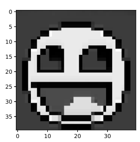
$K = 500,$ $s = 100$



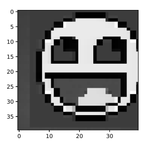
$K = 500,$ $s = 200$



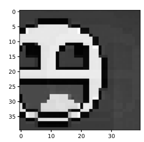
$K = 1000,$ $s = 1$



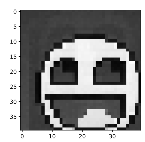
$K = 1000,$ $s = 2$



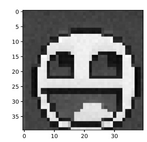
$K = 1000,$ $s = 5$



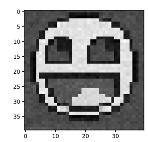
$K = 1000,$ $s = 10$



$K = 1000,$ $s = 50$

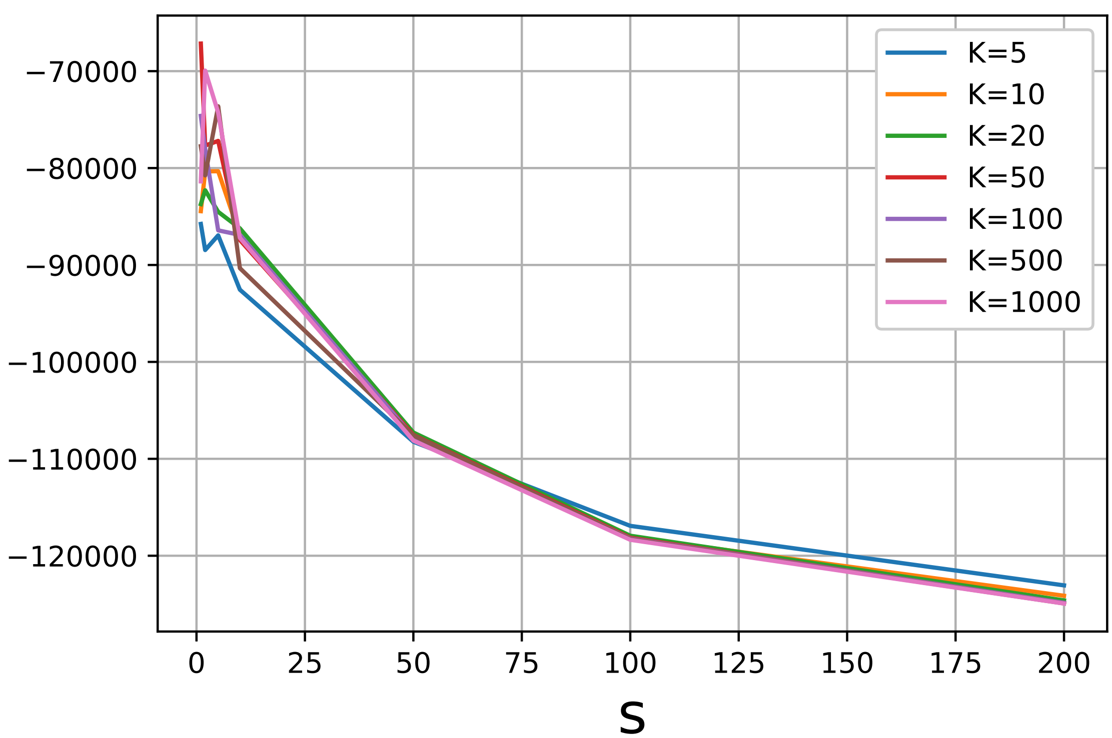


$K = 1000,$ $s = 100$



$K = 1000,$ $s = 200$

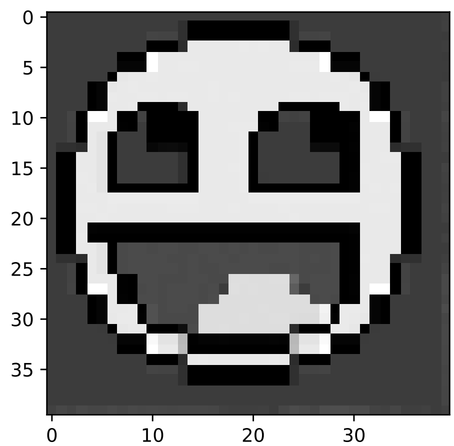
Если не учитывать такие ограничения адекватности дисперсии как максимальное возможное значение яркости пикселя (256), то наблюдаемые результаты вполне логичны: любой аддитивный независимый одинаково распределенный шум можно подавить имея достаточно большую для этого выборку. Убедимся в падении значения $L(q, \boldsymbol{\theta}, \boldsymbol{A})$ от уровня шума:



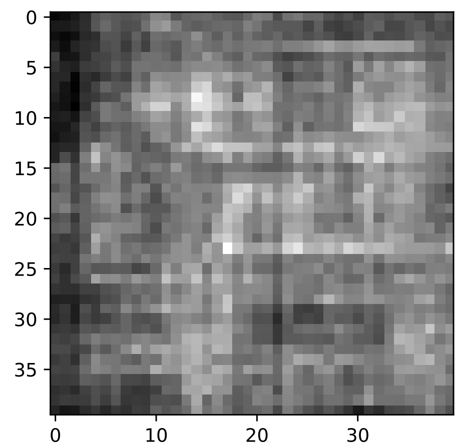
Пункт 3

Сравните качество и время работы EM и hard EM на сгенерированных данных. Как Вы думаете, почему разница в результатах работы так заметна?

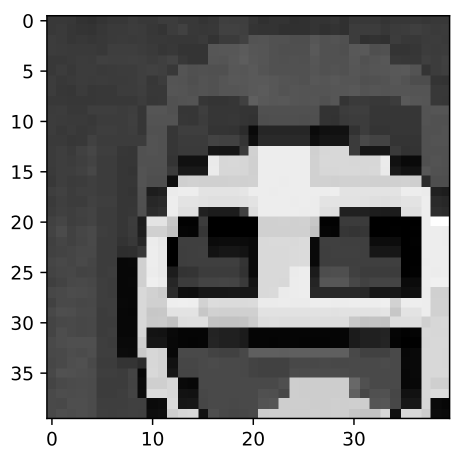
Ну чтож, посмотрим (фиксируем размер выборки и будем менять уровень шума):



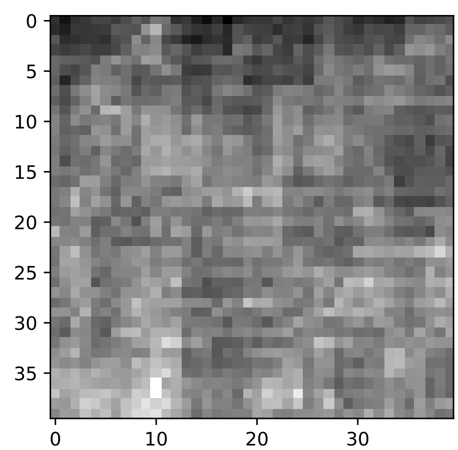
$K = 50,\ s = 1$, full



$K = 50,\ s = 1$, hard

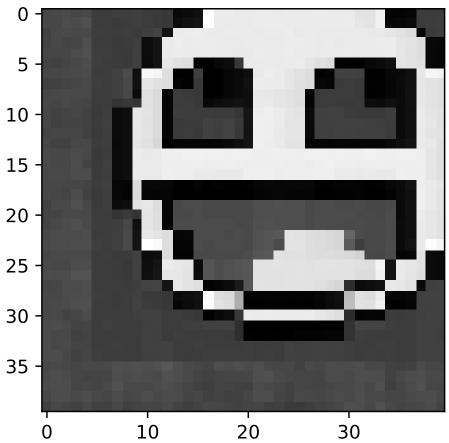


$K = 50,\ s = 2$, full

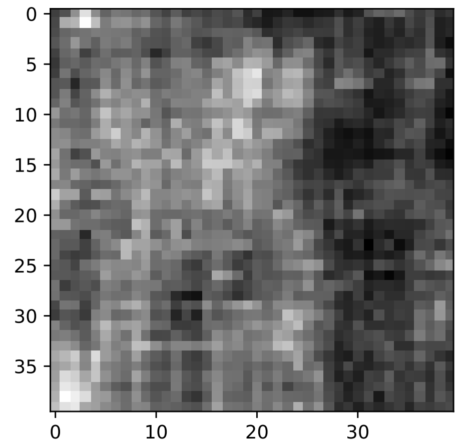


$K = 50,\ s = 2$, hard

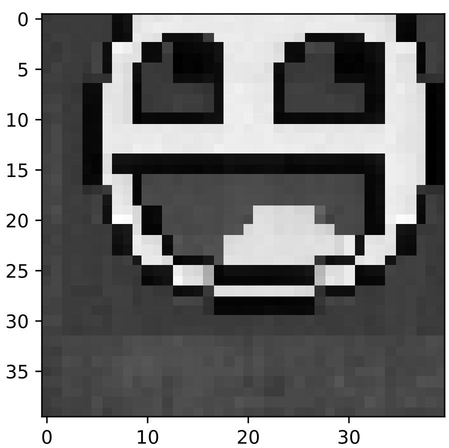
Либо в код закралась ошибка (но я вроде проверял), либо недостаточно большая выборка. Есть ощущение, что основная проблема hard-EM – уменьшение данных, по которым ведется усреднение для рассчета s в $h \cdot w$ раз. Этим и вызвано неспособность восстановить лицо при заданных параметрах. Было бы время, то я бы попробовал провести эксперимент с $K = 2000$ и $s = 10$.



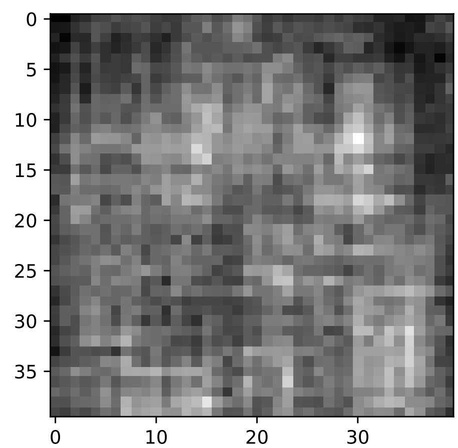
$K = 50,\ s = 5$, full



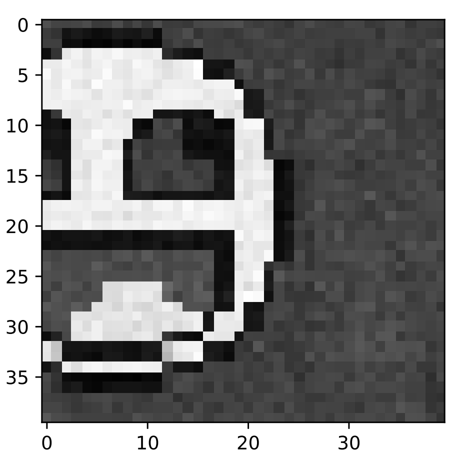
$K = 50,\ s = 5$, hard



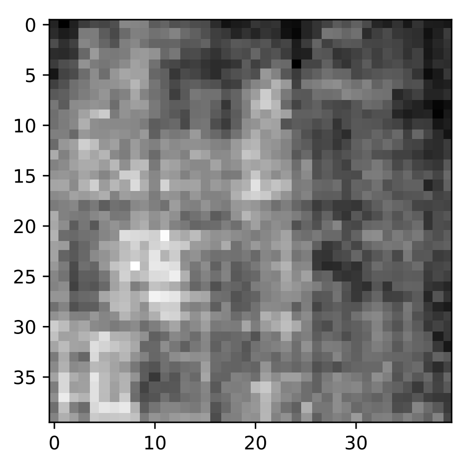
$K = 50,\ s = 10$, full



$K = 50,\ s = 10$, hard



$K = 50,\ s = 50$, full

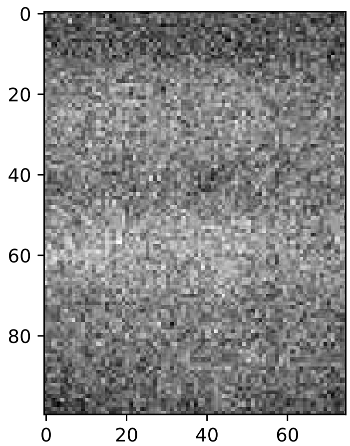


$K = 50,\ s = 50$, hard

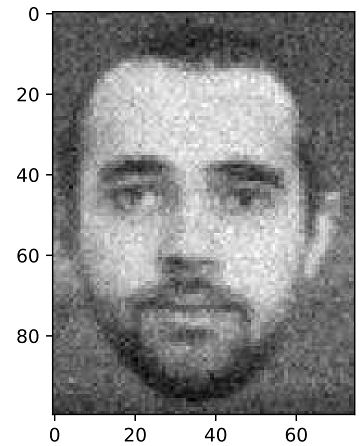
Пункт 4

Примените EM-алгоритм к данным с зашумленными снимками преступника. Приведите результаты работы алгоритма на выборках разного размера.

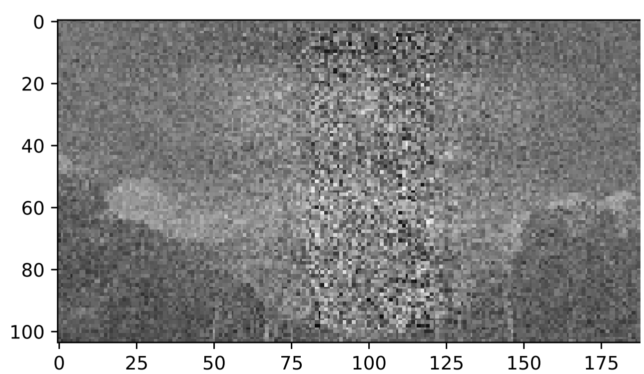
К сожалению я приступил к заданию в тот момент, когда были выложены данные в виду 100 картинок. Так что у меня есть всего лишь 2 результата:



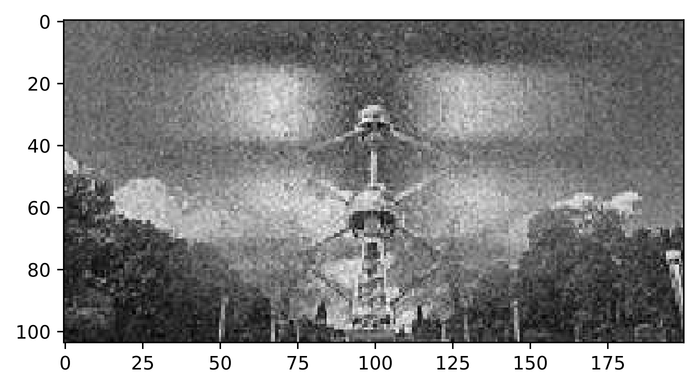
100 картинок



1000 картинок



100 картинок



1000 картинок

Пункт 5

Предложите какую-нибудь модификацию полученного ЕМ алгоритма, которая бы работала на данной задаче качественнее и/или быстрее.

Первое, что пришло в голову: попытаться более граммотно подбирать начальные приближения. В частности, можно найти среднее изображение $\hat{\boldsymbol{X}}$ как простое арифметическое среднее. Далее для каждого $\boldsymbol{X}_k$ рассчитать квадрат разности со средним $\boldsymbol{D}_k$. В нашей задаче гарантируется, что дисперсия у фона и лица одинакова, а значит минимальное значение $\hat{d} = \min_{k,i,j} \boldsymbol{D}_k$ будет хорошей оценкой для s^2. И это значение множно пересчитывать не на каждой итерации, а, например, на каждой 10-ой и в конце. Далее для каждого $\boldsymbol{D}_k$ мы можем найти координаты окон размера $h \times w$ с максимальной суммой и усреднить $\{\boldsymbol{X}_k\}$ отдельно по окнам и отдельно по пространству вокруг них. Хотя это уж очень похоже на М-шаг.