

Московский государственный университет имени М. В. Ломоносова



Факультет Вычислительной Математики и Кибернетики
Кафедра Математических Методов Прогнозирования

**Отчет студента 417 группы
по заданию**

ЕМ-алгоритм для детектива

Выполнил:
студент 417 группы
Бербер Кирилл Андреевич

Москва, 2021

Содержание

1	Введение	2
2	Описание модели	3
3	Формулировка задания	4
4	Теоретические выводы	5
4.1	Апостериорного распределения на координаты лица	5
4.2	Точечные оценок на параметры	5
4.2.1	Параметр A	6
4.2.2	Параметр F	6
4.2.3	Параметр B	6
4.2.4	Параметр s^2	7
4.3	Нижняя оценка на логарифм неполного правдоподобия (ELBO)	7
5	Анализ и эксперименты	9
5.1	Запуск различных размеров	11
5.2	Сравние EM и hard EM	13
5.3	Анализ преступника	15

1 Введение

Леди Маргалотта обратилась к известному детективу Нику Картеру с просьбой расследовать похищение ее любимого песика. Даже огромная страховка не покрывает горя хозяйки... Во время ее поездки в Лондон незнакомец выкрал любимца леди Маргалотты. Камеры видеонаблюдения, зафиксировавшие вора оказались плохими помощниками, т.к. подверглись воздействию направленного электро-магнитного постановщика помех. Тем не менее, это единственная зацепка. Под подозрением группа Байесовских методов из России.

Нику Картеру необходимо обработать снимки и восстановить лицо вора. Известно, что на разных фотографиях оно расположено в случайных координатах на неподвижном фоне. К счастью, видеолекции по курсу БММО уже выложены в сеть и пройдя его Ник Картер смог решить задачу и, с помощью ЕМ-алгоритма, установить личность вора. Его ждал сюрприз...

В задании необходимо помочь детективу изобличить преступника, то есть выделить лицо преступника со снимков фотокамеры. Данные фотокамеры поступали порциями, постепенно увеличивая количество снимков.

Пример зашумленной фотографии показан на рисунке 1.

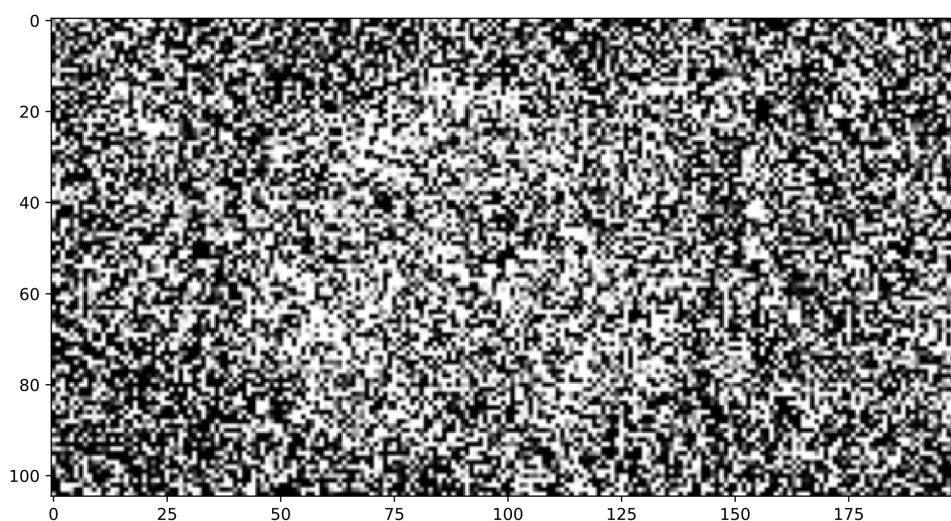


Рис. 1: Пример зашумленного изображения с камер видеонаблюдения

2 Описание модели

Дана выборка $X = \{X_k\}_{k=1}^K$ 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 равны 100 и 66 соответственно.

Макет изображения показан на рисунке 2 .

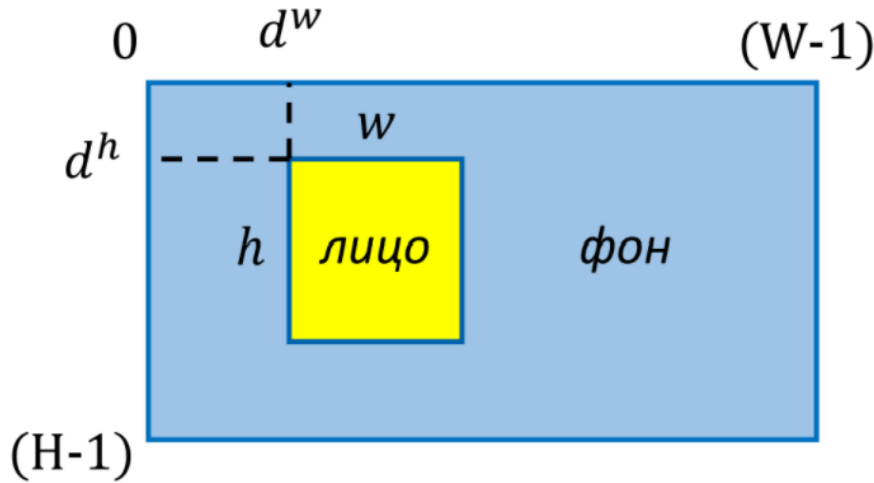


Рис. 2: макет данных без шума

Введем следующие обозначения:

- $X_k(i, j)$ — пиксель k -ого изображения;
- $B \in R^{H \times W}$ — изображение чистого фона без лица преступника, $B(i, j)$ — пиксель этого изображения;
- $F \in R^{h \times w}$ — изображение лица преступника, $F(i, j)$ — пиксель этого изображения;
- $d_k = (d_k^h, d_k^w)$ — координаты верхнего левого угла изображения лица на k -ом изображении (d_k^h по вертикали, d_k^w — по горизонтали), $d = (d_1, \dots, d_K)$ — набор координат для всех изображений выборки.

Также будем считать шум на изображении независимым для каждого пикселя и принадлежащим нормальному распределению $N(0, s^2)$, где s — стандартное откло-

нение. Таким образом для одного изображения имеем:

$$p(X_k|d_k, \theta) = \prod_{i,j} \begin{cases} N(X_k(i, j)|F((i - d_k^h, j - d_k^w), s^2)) & , (i, j) \in \text{FaceArea}(d_k) \\ N(X_k(i, j)|B((i, j), s^2)) & , \text{ иначе} \end{cases}$$

где $\theta = B, F, s^2$, $\text{faceArea}(d_k) = \{(i, j) | d_k^h \leq i \leq d_k^h + h - 1, d_k^w \leq j \leq d_k^w + w - 1\}$.

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

$$p(d_k|A) = A(d_k^h, d_k^w), \sum_{i,j} A(i, j) = 1$$

, где $A(i, j)$ элемент матрицы A .

В итоге имеем следующую совместную вероятностную модель:

$$p(X, d|\theta, A) = \prod_k p(X_k|d_k, \theta) p(d_k|A).$$

3 Формулировка задания

Требуется решить задачу

$$p(X|\theta, A) \rightarrow \max_{\theta, A}$$

Для этого предлагается воспользоваться ЕМ-алгоритмом, то есть перейти к следующей задаче оптимизации нижней оценки на логарифм неполного правдоподобия:

$$\mathfrak{L}(q, \theta, d, A) = E_{q(d)} \log p(X, d|\theta, A) - E_{q(d)} \log q(d) \rightarrow \max_{q, \theta, A}$$

На Е-шаге вычисляется оценка на апостериорное распределение на координаты лица на изображениях:

$$q(d) = p(d|X, \theta, A) = \prod_k p(d_k|X_k, \theta, A),$$

а на М-шаге вычисляется точечная оценка на параметры θ, A :

$$E_{q(d)} \log p(X, d|\theta, A) \rightarrow \max_{\theta, A}$$

Также далее будет рассматриваться упрощенный вариант ЕМ-алгоритма, (hard EM). В нем после Е шага берется не все апостериорное распределение на координаты лица на изображениях, а только МАР оценка на эти координаты (после Е шага $q(d)$ преобразовывают так, что для каждого изображения X_k оценка $q(d_k)$ принимает

значение 1 только в одной точке — точке аргмаксимума апостериорного распределения $p(d_k|X_k, \theta, A)$). При выполнении итераций ЕМ алгоритма необходимо следить за значением оптимизируемого функционала $\mathfrak{L}(q, \theta, A)$.

4 Теоретические выводы

Вывести формулы для подсчета следующих величин:

1. апостериорного распределения на координаты лица на изображениях $p(d_k|X_k, \theta, A)$ на Е-шаге;
2. точечных оценок на параметры $A, \theta = \{F, B, s^2\}$ на М-шаге для ЕМ и MAP-ЕМ алгоритмов
3. нижней оценке на логарифм неполного правдоподобия $\mathfrak{L}(q, \theta, A)$.

4.1 Апостериорного распределения на координаты лица

$$\begin{aligned}
q(d) &= p(d|X, \theta, A) = \prod_k p(d_k|X_k, \theta, A), \\
p(d_k|X_k, \theta, A) &= \frac{p(X_k, d_k|\theta, A)}{p(X_k|\theta, A)} = \frac{p(X_k|d_k, \theta, A) \cdot p(d_k|\theta, A)}{p(X_k|A)} = \frac{p(X_k|d_k, \theta, A) \cdot p(d_k|\theta, A)}{\sum_{d_k} (p(X_k|d_k, \theta, A) \cdot p(d_k|A))} = \\
&= \frac{\frac{1}{s^{H \cdot W} \sqrt{(2\pi)^{H \cdot W}}} \exp \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i - d_k^h, j - d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) \cdot A(d_k^h, d_k^w)}{\sum_{d_k} \frac{1}{s^{H \cdot W} \sqrt{(2\pi)^{H \cdot W}}} \exp \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i - d_k^h, j - d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) \cdot A(d_k^h, d_k^w)} = \\
&= \frac{\exp \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i - d_k^h, j - d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) \cdot A(d_k^h, d_k^w)}{\sum_{d_k} \exp \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i - d_k^h, j - d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) \cdot A(d_k^h, d_k^w)}
\end{aligned}$$

4.2 Точечные оценок на параметры

$$\begin{aligned}
E_{q(d)} \log p(X, d|\theta, A) &= \sum_k \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \log(p(d_k|\theta) p(X_k|d_k, \theta)) = \\
&= \sum_k \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \log(A(d_k^h, d_k^w) \frac{1}{s^{H \cdot W} \sqrt{(2\pi)^{H \cdot W}}}) \cdot \\
&\quad \cdot \exp \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i - d_k^h, j - d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) =
\end{aligned}$$

$$\begin{aligned}
&= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \log(A(d_k^h, d_k^w)) - \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \frac{HW}{2} \log(2\pi) - \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \frac{HW}{2} \log(s^2) - \\
&\sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in F} \frac{(X_k(i,j) - F(i-d_k^h, j-d_k^w))^2}{2s^2} - \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in B} \frac{(X_k(i,j) - B(i,j))^2}{2s^2} = L
\end{aligned}$$

будем дифференцировать это выражение по параметрам. Видно, что от каждого параметра зависят не все слагаемые.

4.2.1 Параметр A.

$$\begin{aligned}
\frac{\partial L}{\partial A(i,j)} = 0 &= \sum_k^K q_k(d_k) \frac{1}{A(i,j)} - \lambda \Rightarrow \lambda A(i,j) = \sum_k^K q_k(d_k) \\
\lambda &= \sum_k^K 1 = K \Rightarrow \\
A(i,j) &= \sum_k^K \frac{q_k(d_k)}{K}
\end{aligned}$$

4.2.2 Параметр F.

$$\begin{aligned}
\frac{\partial L}{\partial(i,j)} &= 2 \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in F} \frac{X_k(i,j) - F(i-d_k^h, j-d_k^w)}{2s^2} = 0 \Rightarrow \\
\sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) F(i,j) \frac{1}{s^2} &= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) X_k(i+d_k^h, j+d_k^w) \frac{1}{s^2} \\
\sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) F(i,j) &= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) X_k(i+d_k^h, j+d_k^w) \\
KF(i,j) &= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) X_k(i+d_k^h, j+d_k^w) \Rightarrow \\
F(i,j) &= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} \frac{q_k(d_k) X_k(i+d_k^h, j+d_k^w)}{K}
\end{aligned}$$

4.2.3 Параметр B.

$$\frac{\partial L}{\partial B(i,j)} = 2 \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in B} \frac{X_k(i,j) - B(i,j)}{2s^2} = 0 \Rightarrow$$

$$\sum_k^K \sum_{i \in B} \sum_{j \in B} B(i, j) \frac{1}{s^2} = \sum_k^K \sum_{i \in B} \sum_{j \in B} q_k(d_k) X_k(i, j) \frac{1}{s^2}$$

Мы знаем, что фон там, где нет области лица. Применим эти знания для вывода. Вычтем то что накладывается, учитывая размеры изображения (суммирование не должно выходить за пределы)

$$B(i, j) = \frac{\sum_k^K X_k(i, j) - \sum_k^K \sum_{\substack{\min(i, H-h) \\ \max(0, i-h+1)}}^{\min(j, W-w) \\ \max(0, j-w+1)} q_k(d_k) X_k(i, j)}{K - \sum_k^K \sum_{\substack{\min(i, H-h) \\ d_k^h = \max(0, i-h+1)}}^{\min(j, W-w) \\ \max(0, j-w+1)} q_k(d_k)}$$

4.2.4 Параметр s^2 .

$$\frac{\partial L}{\partial s^2} = \frac{K \cdot WH}{s} - \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in F} (X_k(i, j) - F(i - d_k^h, j - d_k^w))^2 -$$

$$\sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \sum_{i,j \in B} (X_k(i, j) - B(i, j))^2 = 0 \Rightarrow$$

$$s^2 = \frac{1}{K \cdot WH} \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \left(\sum_{i,j \in F} (X_k(i, j) - F(i - d_k^h, j - d_k^w))^2 + \sum_{i,j \in B} (X_k(i, j) - B(i, j))^2 \right)$$

4.3 Нижняя оценка на логарифм неполного правдоподобия (ELBO)

$$\mathfrak{L}(q, \theta, d, A) = E_{q(d)} \log p(X, d | \theta, A) - E_{q(d)} \log q(d) = E_{q(d)} \sum_k^K \log p(X_k, d_k | \theta, A) - E_{q(d)} \log q(d) =$$

$$= \sum_k^K E_{q(d)} \log p(X_k | d_k, \theta, A) \cdot p(d_k | \theta, A) - \sum_k^K E_{q(d)} \log q_k(d_k) =$$

$$= \sum_k^K E_{q(d)} \log p(X_k | d_k, \theta, A) + \sum_k^K \log p(d_k | \theta, A) - \sum_k^K E_{q(d)} \log q_k(d_k) =$$

$$= \sum_k^K E_{q(d)} \log p(X_k | d_k, \theta, A) + \sum_k^K \log p(d_k | \theta, A) - \sum_k^K E_{q(d)} \log q_k(d_k) =$$

$$\begin{aligned}
&= (-1) \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \frac{HW}{2} \log(2\pi s^2) + \\
&+ \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \left(\sum_{i,j \in F} -\frac{(X_k(i,j) - F(i-d_k^h, j-d_k^w))^2}{2s^2} + \sum_{i,j \in B} -\frac{(X_k(i,j) - B(i,j))^2}{2s^2} \right) + \\
&+ \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \log(A(d_k^h, d_k^w)) - \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \log(q_k(d_k^h, d_k^w)) = \\
&= \sum_k^K \sum_{d_k^h=0}^{H-h} \sum_{d_k^w=0}^{W-w} q_k(d_k) \left(-\frac{HW}{2} \log(2\pi s^2) + \frac{-\sum_{i,j \in F} (X_k(i,j) - F(\dots))^2 - \sum_{i,j \in B} (X_k(i,j) - B(i,j))^2}{2s^2} \right) + \\
&\quad + \log(A(d_k^h, d_k^w)) - \log(q_k(d_k^h, d_k^w))
\end{aligned}$$

Причем при использовании hard-ЕМ алгоритма происходит упрощение этого выражения:

$$\mathfrak{L}(q, \theta, d, A) = E_{q(d)} \log p(X, d | \theta, A) - E_{q(d)} \log q(d) \rightarrow \max_{q, \theta, A}$$

$$\begin{aligned}
\mathfrak{L}(q, \theta, d, A) &= E_{q(d)} \log p(X, d | \theta, A) - E_{q(d)} \log q(d) = \\
&= \sum_k^K E_{q(d)} \log p(X_k | d_k, \theta, A) + \sum_k^K \log p(d_k | \theta, A) - \sum_k^K E_{q(d)} \log q_k(d_k)
\end{aligned}$$

В этом случае последнее слагаемое обращается в ноль (там можно заметить энтропию врожденного распределения q). Во всех суммах будет участвовать только одно слагаемое и в результате получается:

$$-\left(\frac{HW}{2} \log(2\pi s^2) - \sum_k^K \left(\frac{\sum_{i,j \in F} (X_k(i,j) - F(i-d_k^h, j-d_k^w))^2}{2s^2} + \frac{\sum_{i,j \in B} (X_k(i,j) - B(i,j))^2}{2s^2} \right) \right) + \log(A(d_k^h, d_k^w))$$

5 Анализ и эксперименты

Для проведения экспериментов была составлена специальная картинка (Face) которую хотелось отгадывать и некоторые фоны разных размеров. (Рис. 3), (Рис. 4)

После генерации картинки и фона был протестирован алгоритм, написанный ранее.

Запуск алгоритма из различных начальных приближений действительно стоит запускать. В некоторых случаях алгоритм сходился в локальный максимум и результат получался абсолютно неинтерпретируемым. Так же это влияло на время работы алгоритма. Сравнение работы двух алгоритмов из разных начальных приближений.

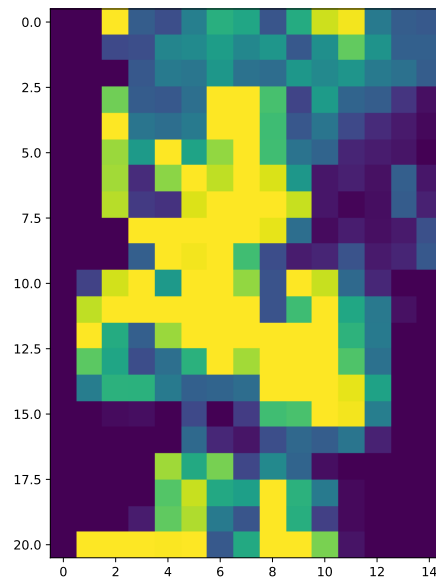


Рис. 3: Изображение Face

	количество слоев	$L(q, \theta, A)$	время работы, с
random	150	-90733	57
zeros	150	-90739	15

Как видно из эксперимента, изменение начальных данных привело к улучшению времени работы, однако стоит посмотреть на выходные данные (Face) и становится понятно, что алгоритм сошелся не туда и результат получился неинтерпретируемым (Рис. 5).

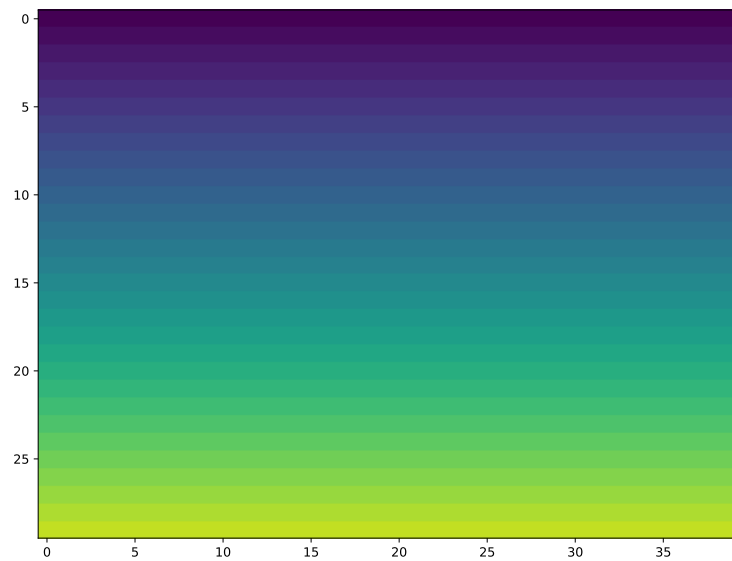


Рис. 4: изображение background (пример)

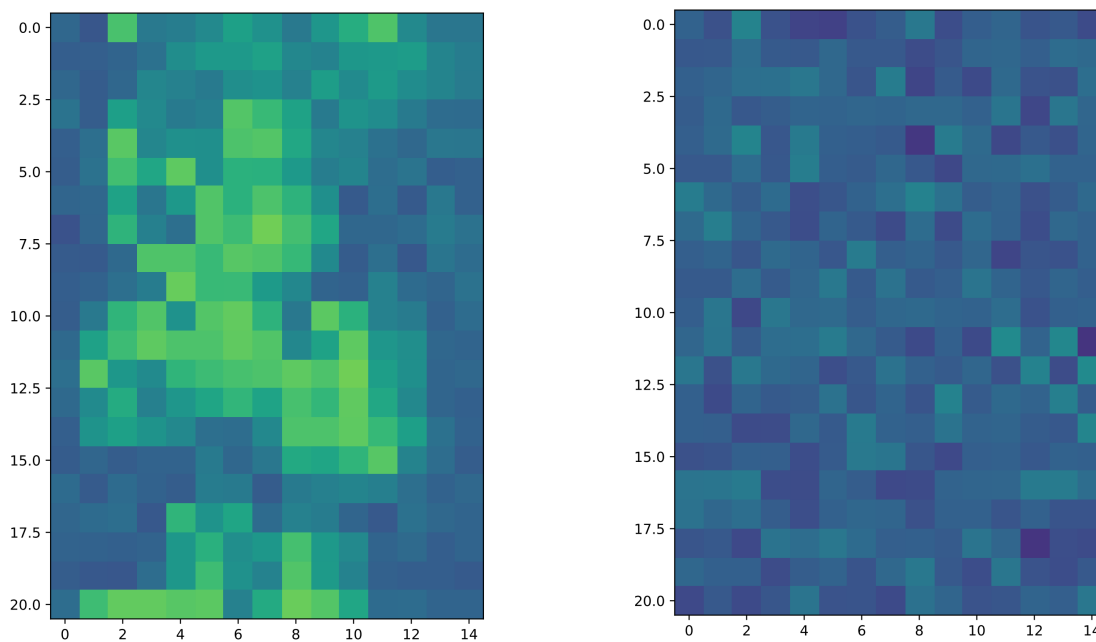


Рис. 5: Сравнение результатов работы на разных начальных приближениях

Алгоритм в одном из случаев действительно правильно проявил изображение. Но все же стоит отметить, что таких случаев сходимости не туда было не очень много. Однако для более корректного результата стоит запускать алгоритм несколько раз.

5.1 Запуск различных размеров

Далее происходил запуск алгоритма на разном количестве слоев.

При нормировку на количество слоев выборки ELBO практически не менялось (те росло линейно относительно количества слоев). Качество изображений с увеличением количество слоев тоже возрастало.

Остановимся подробнее на шуме. Выведем несколько изображений при различном шуме на 100 слоях ЕМ алгоритма, чтобы понять с какого числа картинки начинают быть неузнаваемыми. Также убедимся, что чем меньше s , тем четче изображение

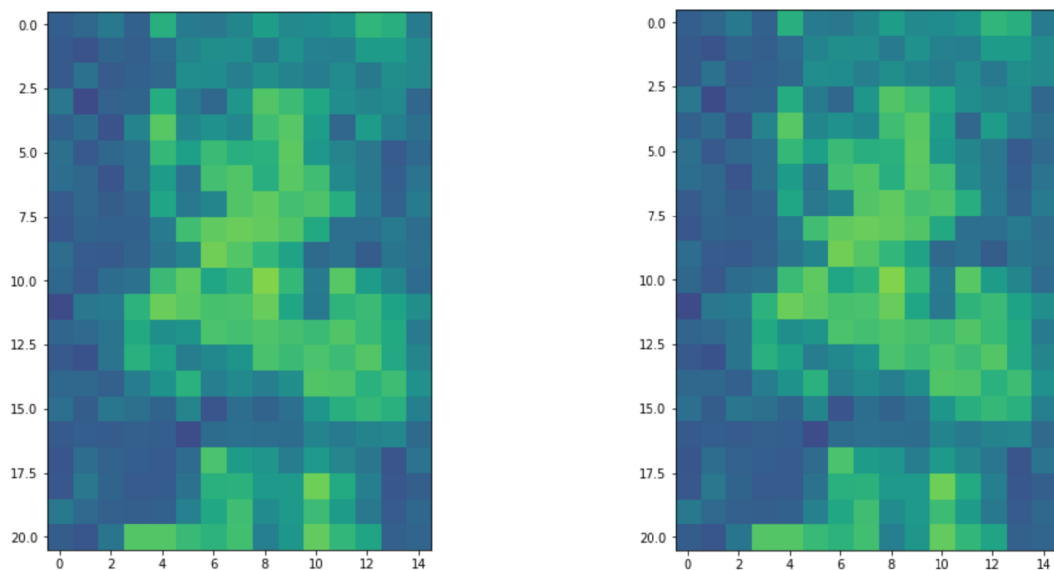


Рис. 6: Результаты, 100 слоев, $s = 0.1, 0.5$

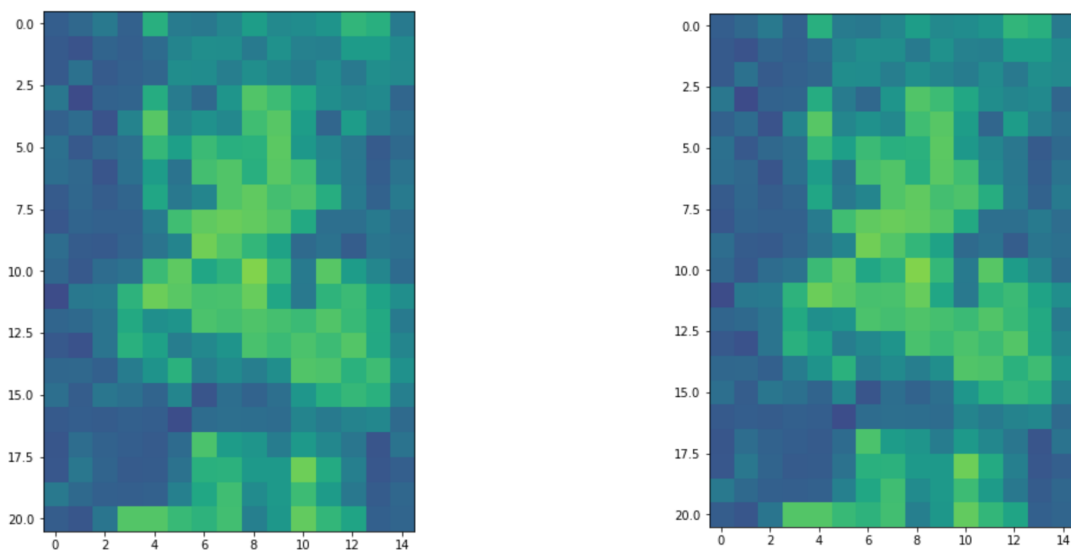


Рис. 7: Результаты, 100 слоев, $s = 0.8, 1.05$

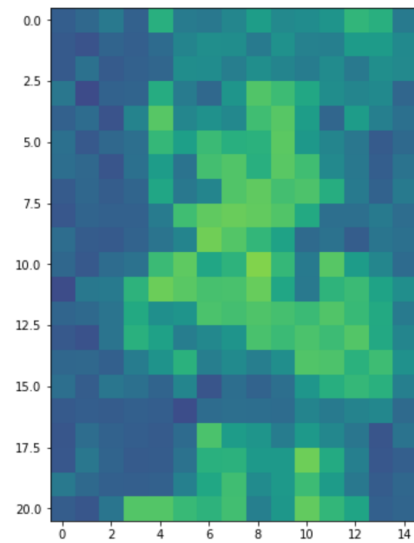
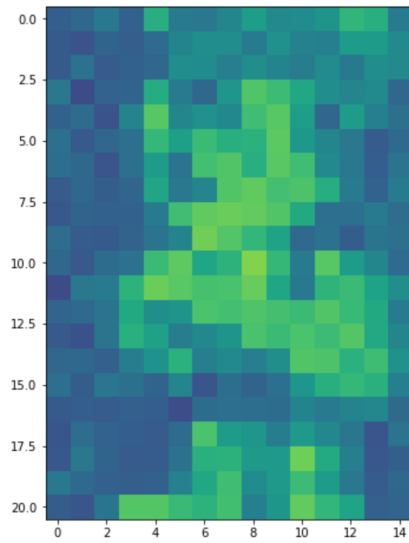


Рис. 8: Результаты, 100 слоев, $s = 1.5, 2$

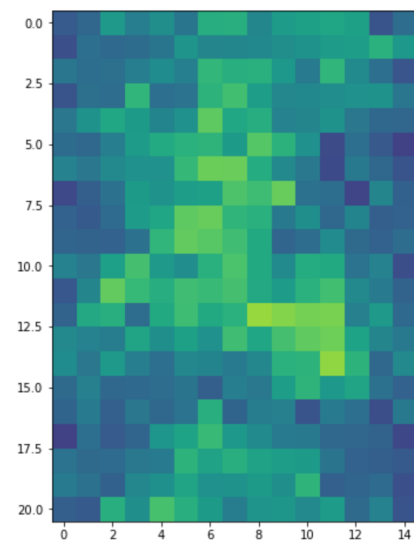
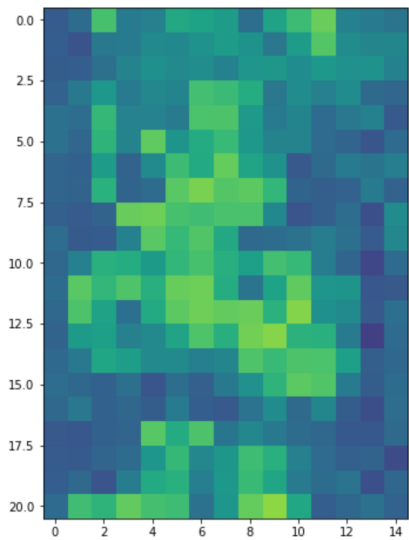


Рис. 9: Результаты, 100 слоев, $s = 3, 10$

Видно, что при небольших показателях s (< 1) все обрабатывается неплохо. Однако при увеличении s картинка начинает немного больше расплываться при $s = 2$ и более изображение становится нечитаемым.

5.2 Сравние ЕМ и hard ЕМ

В данном пункте необходимо сравнить работу ЕМ и hard ЕМ алгоритмов. При запуске было видно, что hard ЕМ алгоритм работает куда быстрее, что видно из графика.

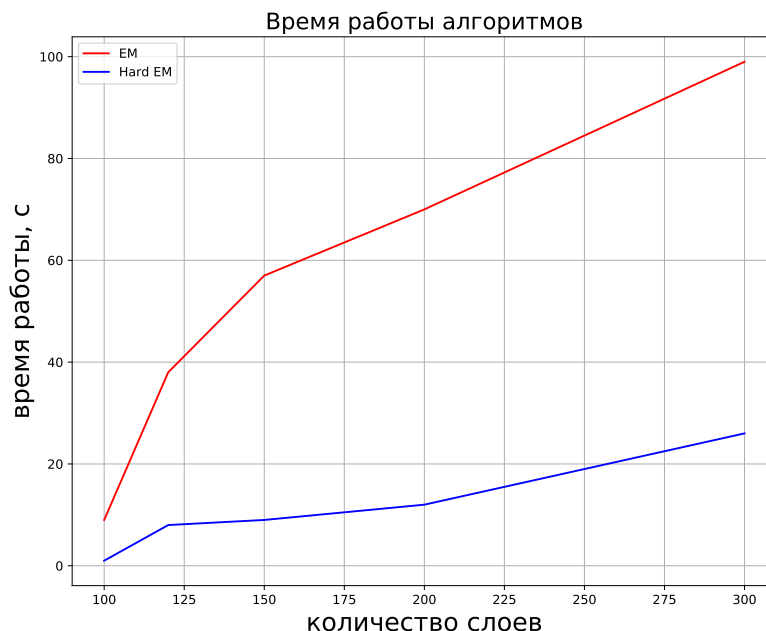


Рис. 10: Сравнение времени работы алгоритмов

Однако результаты, которые получались при работе hard ЕМ начинают быть интерпретируемыми только при достаточно большом количестве слоев, что логично. Ведь там в матрице q берется $\arg\max$. Стоит отметить, что если слоев очень много, то hard-ЕМ алгоритм показывает хорошие результаты, и не обязательно пользоваться первым. Посмотрим на результат на графиках ниже

На большом количестве слоев Hard-ЕМ показал силуэт ёжика.

Таким образом можно сделать вывод, что при небольшом объеме данных стоит пользоваться ЕМ алгоритмом, но если слоев достаточно количество, то для уменьшения времени работы можно использовать и hard-ЕМ. Преступника можно найти и с помощью второго метода.

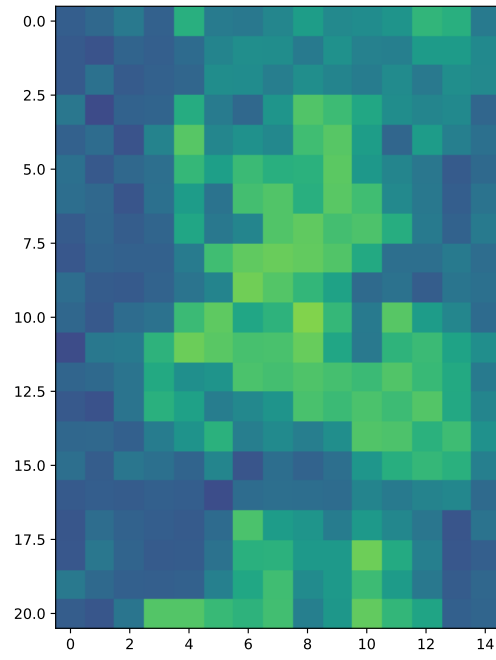


Рис. 11: EM алгоритм, 100 слоев

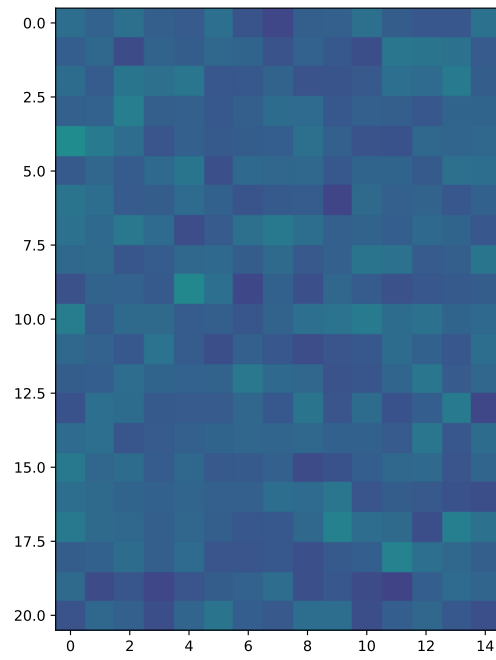


Рис. 12: hard-EM, 300 слоев

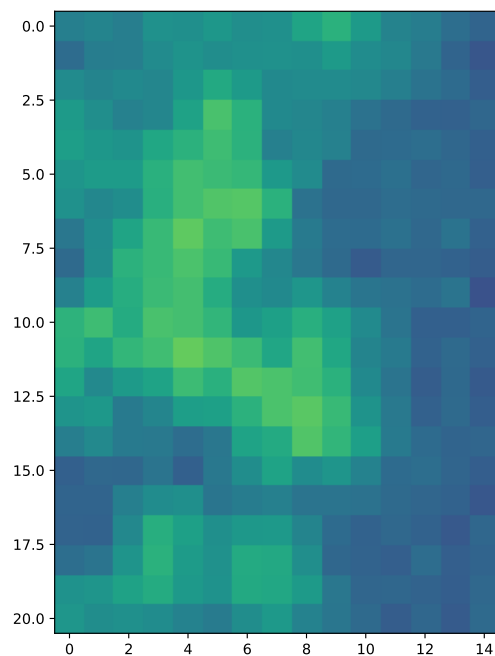


Рис. 13: hard-EM, 400 слоев

5.3 Анализ преступника

В данном разделе надо было применить работу алгоритма на данных, полученных от преподавателей и понять что изображено на изображении (место и человека).

Производились запуски на нескольких наборах данных, однако достойные результаты получались только на выборках большого размера. Ниже приведены примеры работы алгоритма на размере 50 и 500 слоев.

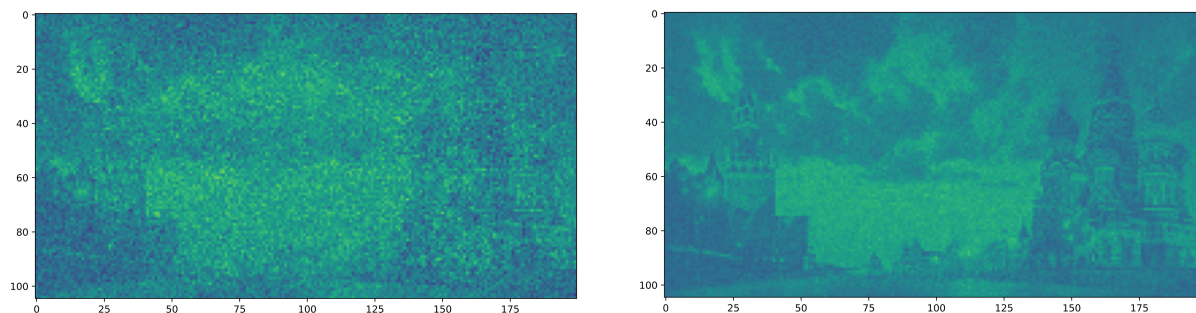


Рис. 14: Сравнение фона, количество слоев 50 и 500

После выполнения алгоритма Красная Площадь видна и при 50 и при 500 итерациях, но лишь на 500 мы можем действительно сказать, что преступником был Максим! С чем мы и поздравляем детектива.

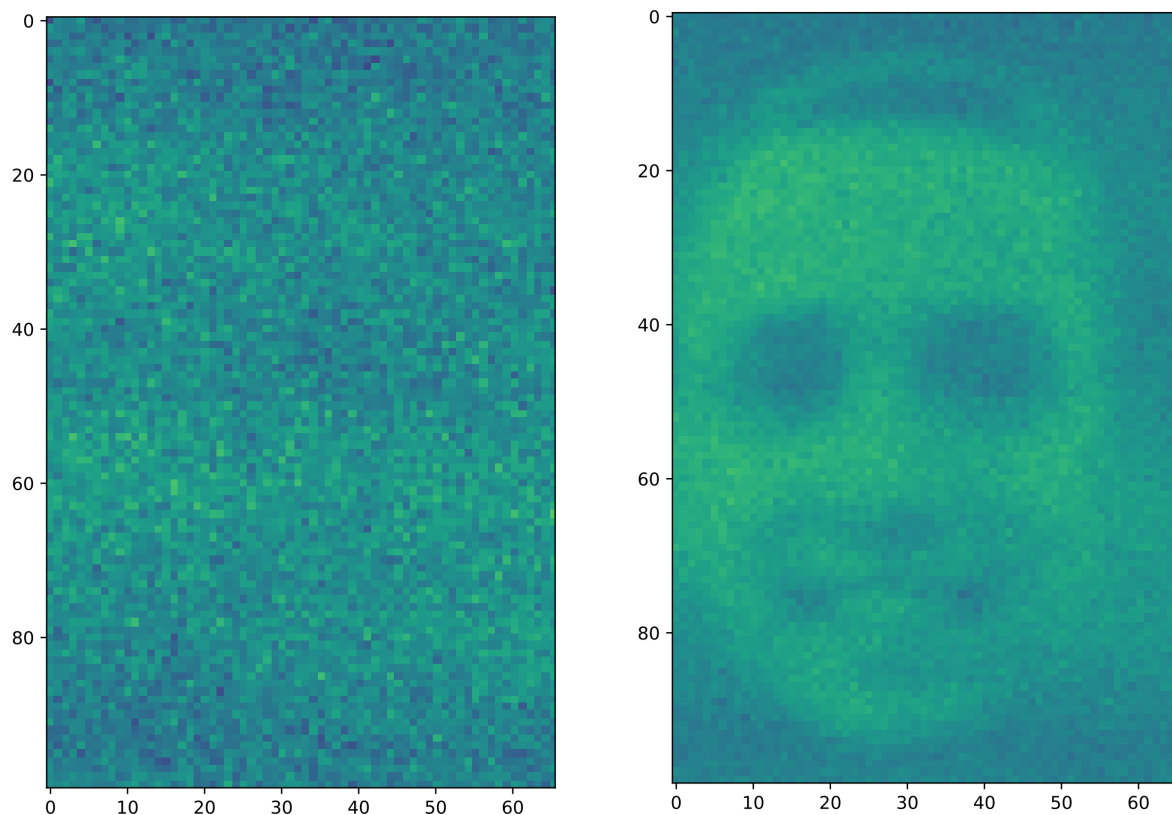


Рис. 15: Сравнение лица преступника, количество слоев 50 и 500

Алгоритм действительно можно было улучшить, не считая ELBO на каждой итерации цикла, потому что это не самая быстро работающая функция, потому что в своем расчете она использует `calculate log probability`. Так же можно вспомнить, что на M шаге происходит максимизация параметров. Однако Нам совсем необязательно находить эту оценку точно, достаточно лишь улучшить ее по сравнению с предыдущим шагом. Вероятно этот момент может ускорить выполнение M шага и сделать алгоритм быстрее и лучше. Потери качества в этом случае происходить не будет, возможно лишь увеличится количество итераций. Этот способ не гарантирует ускорение по времени, но возможно в некоторых случаях время работы действительно уменьшится.