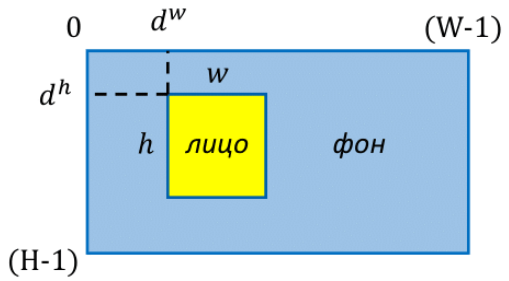
Задание 2. EM алгоритм для детектива

Курс: Байесовские методы в машинном обучении, осень 2021

1 Формулировка задания

Дана выборка $X = \{X_k\}_{k=1}^K$ сильно зашумленных черно-белых изображений размера $H \times W$ из файла формата `.npy` (сохраненный `numpy array` размера $H \times W \times K$). Каждое из этих изображений содержит один и тот же неподвижный фон и лицо преступника размерами $h \times w$ в неизвестных координатах, при этом лицо попадает в любое изображение целиком. Значения h, w указаны в `anytask`.



B - изображение фона $\in \mathbb{R}^{H \times W}$
F - изображение лица $\in \mathbb{R}^{h \times w}$
ε - независимый шум в каждом пикселе изображения $\sim \mathcal{N}(\varepsilon|0, s^2)$
X_k - k-е итоговое изображение $\in \mathbb{R}^{H \times W}$
$d_k = (d_k^h, d_k^w)$ - координаты левого верхнего угла изображения лица преступника F на итоговом изображении X_k.

Значит, каждый пиксель

$$X_k[i,j] = \begin{cases} F[i - d_k^h, j - d_k^w] + \varepsilon, \text{ если } [i,j] \in F_{area} \\ B[i,j] + \varepsilon, \text{ иначе} \end{cases} \qquad F_{area} = [d_k^h : d_k^h + h) \times [d_k^w : d_k^w + w) \tag{1}$$

Значит, поскольку все пиксели независимы:

$$p(X_k \mid d_k, \theta) = \prod_{ij} \begin{cases} \mathcal{N}(X_k[i,j] \mid F[i - d_k^h, j - d_k^w], s^2), & \text{если } [i,j] \in F_{area} \\ \mathcal{N}(X_k[i,j] \mid B[i,j], s^2), & \text{иначе} \end{cases}, \quad \theta = \{B, F, s^2\} \tag{2}$$

Априорное распределение d_k задано с помощью матрицы вероятностей A:

$$p(d_k \mid A) = A[d_k^h, d_k^w] \tag{3}$$

В итоге вероятностная модель:

$$p(X, d \mid \theta, A) = \prod_k p(X_k \mid d_k, \theta) p(d_k \mid A) \tag{4}$$

Требуется решить задачу:

$$p(X \mid \theta, A) \to \max_{\theta, A} \tag{5}$$

2 Теоретическая часть

2.1 Метод решения

Воспользуемся EM-алгоритмом, то есть перейдем к задаче оптимизации нижней оценки на логарифм неполного правдоподобия:

$$\mathcal{L}(q, \theta, A) = \mathbb{E}_{q(d)}[\log p(X, d \mid \theta, A)] - \mathbb{E}_{q(d)}[\log q(d)] \to \max_{q, \theta, A} \tag{6}$$

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

$$q(d) = p(d \mid X, \theta, A) = \prod_k p(d_k \mid X_k, \theta, A) \tag{7}$$

На **M-шаге** вычисляется точечная оценка на параметры θ, A:

$$\mathbb{E}_{q(d)}[\log p(X, d \mid \theta, A)] \to \max_{\theta, A} \tag{8}$$

Также далее будет рассматриваться упрощенный вариант EM-алгоритма, который называется hard EM. В нем после E шага берется не все апостериорное распределение на координаты лица на изображениях, а только MAP оценка на эти координаты.

2.2 Теоретические задания

Вывести формулы для подсчета следующих величин:

1. Апостериорного распределения на координаты лица на изображениях $p(d_k \mid X_k, \theta, A)$ на E-шаге:

   Решение:

$$p(X_k \mid d_k, \theta) = \prod_{[i,j] \in F_{area}(d_k)} \mathcal{N}(X_k[i,j] \mid F[i - d_k^h, j - d_k^w], s^2) \prod_{[i,j] \notin F_{area}(d_k)} \mathcal{N}(X_k[i,j] \mid B[i,j], s^2) =$$
$$= \sqrt{2\pi s^2}^{-hw} \sqrt{2\pi s^2}^{-(H-h)(W-w)} e^{-\frac{\sum\limits_{[i,j] \in F_{area}(d_k)} (X_k[i,j] - F[i-d_k^h, j-d_k^w])^2 \quad + \sum\limits_{[i,j] \notin F_{area}(d_k)} (X_k[i,j] - B[i,j])^2}{2s^2}} =$$
$$= \sqrt{2\pi s^2}^{-HW} e^{-\frac{\|X_k - Y_k\|^2}{2s^2}}$$

   $Y_k = Y(d_k, B, F)$ - изображение лица преступника F в позиции d_k на фоне B без шума.

$$p(X_k \mid d_k, \theta) p(d_k \mid A) = \sqrt{2\pi s^2}^{-HW} e^{-\frac{\|X_k - Y_k\|^2}{2s^2}} A[d_k^h, d_k^w]$$
$$\Downarrow$$
$$q_k(d_k) = p(d_k \mid X_k, \theta, A) = \frac{p(X_k \mid d_k, \theta) p(d_k \mid A)}{\sum\limits_{d_k} p(X_k \mid d_k, \theta) p(d_k \mid A)} = \frac{e^{-\frac{\|X_k - Y_k\|^2}{2s^2}} A[d_k^h, d_k^w]}{\sum\limits_{d_k} e^{-\frac{\|X_k - Y_k\|^2}{2s^2}} A[d_k^h, d_k^w]}$$

2. Точечных оценок на параметры $A, \theta = \{F, B, s^2\}$ на M-шаге для EM и MAP-EM алгоритмов (точечные оценки здесь нужно получать именно в таком порядке: A, F, B, s^2)

   Решение:

   EM:

$$\mathbb{E}_{q(d)}[\log p(X, d \mid \theta, A)] = \sum_k \mathbb{E}_{q_k(d_k)}[\log p(X_k, d_k \mid \theta, A)] = \sum_k \mathbb{E}_{q_k(d_k)}[\log p(X_k \mid d_k, \theta) + \log p(d_k \mid A)] =$$
$$= \sum_k \mathbb{E}_{q_k(d_k)}[-HW(\frac{\log 2\pi}{2} - \log s) - \frac{1}{2s^2}\|X_k - Y_k\|^2 + \log A[d_k^h, d_k^w]] =$$
$$= -KHW(\frac{\log 2\pi}{2} - \log s) - \frac{1}{2s^2} \sum_k \mathbb{E}_{q_k(d_k)} \|X_k - Y_k\|^2 + \sum_k \sum_{d_k} q_k(d_k) \log A[d_k] \to \max_{\theta, A}$$

   (a) Минимизируя по A, будем учитывать, что A - матрица вероятностей : $\sum\limits_{i,j} A[i,j] = 1$.

   В таком случае удобно воспользоваться методом множителей Лагранжа: $(9) + \lambda(1 - \sum\limits_{i,j} A[i,j]) \to \max\limits_A$

   Поскольку в (9) от A зависит только $f = \sum\limits_k \sum\limits_{d_k} q_k(d_k) \log A[d_k]$, приходим к задаче:

$$f + \lambda(1 - \sum_{i,j} A[i,j]) \to \max_A$$

   Дифференцируем скаляр по каждому элементу матрицы A и приравниваем к 0:

$$\frac{\partial \dots}{\partial A[d_k]} = \sum_k \frac{q_k(d_k)}{A[d_k]} - \lambda = 0 \Rightarrow A[d_k] = \lambda^{-1} \sum_k q_k(d_k) \Rightarrow \sum_{d_k} A[d_k] = \sum_{d_k} \lambda^{-1} \sum_k q_k(d_k) = \sum_k \lambda^{-1} = 1 \Rightarrow \lambda = K.$$

   Итого: $\boxed{A[d_k] = \frac{\sum\limits_k q_k(d_k)}{K}}$

(b) От F и B в (9) зависит только $f = -\frac{1}{2s^2}\sum\limits_k \mathbb{E}_{q_k(d_k)}\|X_k - Y_k\|^2$ потому как $Y_k = Y_k(d_k, F, B)$

$$\|X_k - Y_k\|^2 = \sum_{[i,j]\in F_{area}(d_k)} (X_k[i,j] - F[i-d_k^h, j-d_k^w])^2 + \sum_{[i,j]\notin F_{area}(d_k)} (X_k[i,j] - B[i,j])^2$$

Значит, $s^2\frac{\partial f}{\partial F[a,b]} = \sum\limits_k \mathbb{E}_{q_k(d_k)}[X_k[a+d_k^h, b+d_k^w] - F[a,b]] = (\sum\limits_k \mathbb{E}_{q_k(d_k)} X_k[a+d_k^h, b+d_k^w]) - KF[a,b] = 0 \Rightarrow$

$$\boxed{F[a,b] = \frac{\sum\limits_k \mathbb{E}_{q_k(d_k)} X_k[a+d_k^h, b+d_k^w]}{K}}$$

Для B: $s^2\frac{\partial f}{\partial B[a,b]} = \sum\limits_k \sum\limits_{\substack{d_k: \\ [a,b]\notin F_{area}(d_k)}} q_k(d_k)[X_k[a,b] - B[a,b]] = \sum\limits_k q'_k[a,b](X_k[a,b] - B[a,b]) = 0 \Rightarrow$

$$\boxed{B[a,b] = \frac{\sum\limits_k X_k[a,b] q'_k[a,b]}{\sum\limits_k q'_k[a,b]}}, \text{ где } q'_k[a,b] = 1 - \sum_{\substack{d_k: \\ [a,b]\in F_{area}(d_k)}} q_k(d_k)$$

(c) Минимизируем по s. От s в (9) зависит только $f = -KHW\log s - \frac{1}{2s^2}\sum_k \mathbb{E}_{q_k(d_k)}\|X_k - Y_k\|^2$

$$\frac{\partial f}{\partial s} = -KHWs^{-1} + s^{-3}\textstyle\sum_k \mathbb{E}_{q_k(d_k)}[\|X_k - Y_k\|^2] = 0 \Rightarrow \boxed{s^2 = \frac{\sum_k \mathbb{E}_{q_k(d_k)}\|X_k - Y_k\|^2}{KHW}}$$

MAP-EM:
Скорректируем формулы выше с учетом, что теперь матрица q_k состоит из нулей везде, кроме d_k^{MAP} - точки максимума апостериорной вероятности.

(a) Для A можно ничего не менять: $\boxed{A[d_k] = \frac{\sum\limits_k q_k(d_k)}{K}}$

(b) $\boxed{F[a,b] = \frac{\sum\limits_k X_k[[a,b]+d_k^{MAP}]}{K}}$

(c) $\boxed{B[a,b] = \frac{\sum\limits_k X_k[a,b]\times[[a,b]\notin F_{area}(d_k^{MAP})]}{\sum\limits_k [[a,b]\notin F_{area}(d_k^{MAP})]}}$

(d) $\boxed{s^2 = \frac{\sum_k \|X_k - Y_k(d_k^{MAP})\|^2}{KHW}}$, где Y_k - фон B с лицом преступника F на позиции d_k^{MAP}

3. Нижней оценке на логарифм неполного правдоподобия $\mathcal{L}(q, \theta, A)$.
Подставим значения, полученные выше.

EM:

$$\mathcal{L}(q,\theta,A) = \mathbb{E}_{q(d)}[\log p(X, d \mid \theta, A) - \log q(d)] = \mathbb{E}_{q(d)}[\log p(X \mid d, \theta) + \log p(d \mid A) - \log q(d)] =$$
$$= \mathbb{E}_{q(d)}\sum_k [\log p(X_k \mid d_k, \theta) + \log p(d_k \mid A) - \log q_k(d_k)] = \sum_k \mathbb{E}_{q_k(d_k)} Q_k[d_k]$$

$\boxed{\mathcal{L}(q,\theta,A) = \sum_k \mathbb{E}_{q_k(d_k)} Q_k[d_k]}$ $Q_k = \log p(X_k \mid d_k, \theta) + \log p(d_k \mid A) - \log q_k(d_k)$

MAP-EM:
Из формулы для $\mathcal{L}(q, \theta, A)$ нужно только убрать $\mathbb{E}_{q(d)}[\log q(d)] = 0$
$\boxed{\mathcal{L}(q,\theta,A) = \sum_k Q_k[d_k^{MAP}]}$ $Q_k = \log p(X_k \mid d_k, \theta) + \log p(d_k \mid A)$

3 Практическая часть

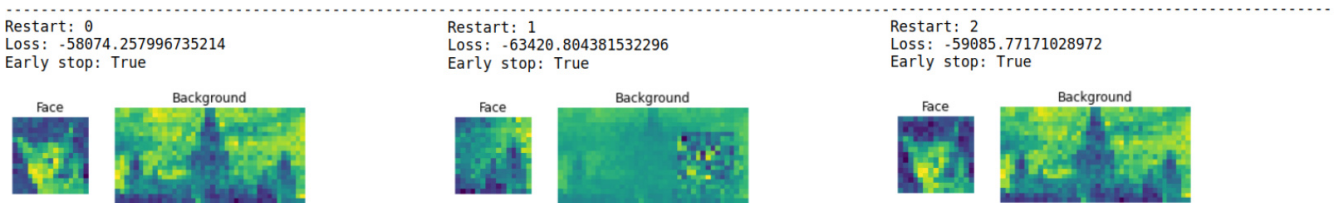
Необходимо реализовать:

1. EM-алгоритм со вспомогательными функциями. В качестве критерия останова использовать следующее условие:
$$\mathcal{L}(q, \theta^{(t+1)}, A^{(t+1)}) - \mathcal{L}(q, \theta^{(t)}, A^{(t)}) < tol,$$
2. Дополнить функции для выполнения M шага и EM алгоритма на случай hard EM алгоритма.
3. Функцию, запускающую EM-алгоритм несколько раз из разных начальных приближений.
4. Для проверки работы алгоритма сгенерировать выборку из небольших зашумленных нормальным шумом черно-белых изображений с одинаковым фоном и каким-то объектом в случайной позиции.

4 Итоги

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?

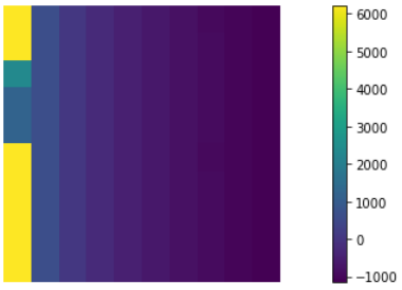
Запустим алгоритм с разными начальными приближениями:



Как видно, начальное приближение (а именно им и отличаются эти три запуска) оказывается важным для результата. Можно было бы предположить, что EM просто не успел сойтись за такое число шагов, однако это не так, и EM для каждого запуска совершал ранюю остановку, так как функционал качества прекращал расти. Поэтому отличие картинок говорит о важности начального приближения. Что довольно логично, потому как у функционала $\mathcal{L}(q, \theta, A)$ не обязательно один максимум.

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

Будем запускать алгоритм с разными значениями K и s с 5 разных начальных прилижений для каждой пары K, s, а затем выбирать лучший из 5. Тепловая карта для значений функционала $\mathcal{L}(q, \theta, A)$:



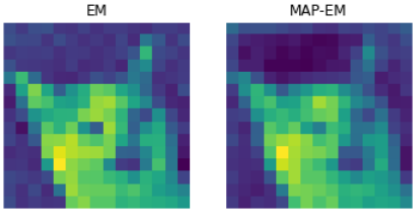
Сами лица для разных K, s. Левый нижний угол - самые маленькие K, s, правый верхний - самые большие.

Видна прямая зависимость между размером выборки K и дисперсией s: чем больше дисперсия, тем больше должна быть выборка.

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

Сравнивать будем лучшие результаты по 5 запускам для каждого значения `use_MAP`:

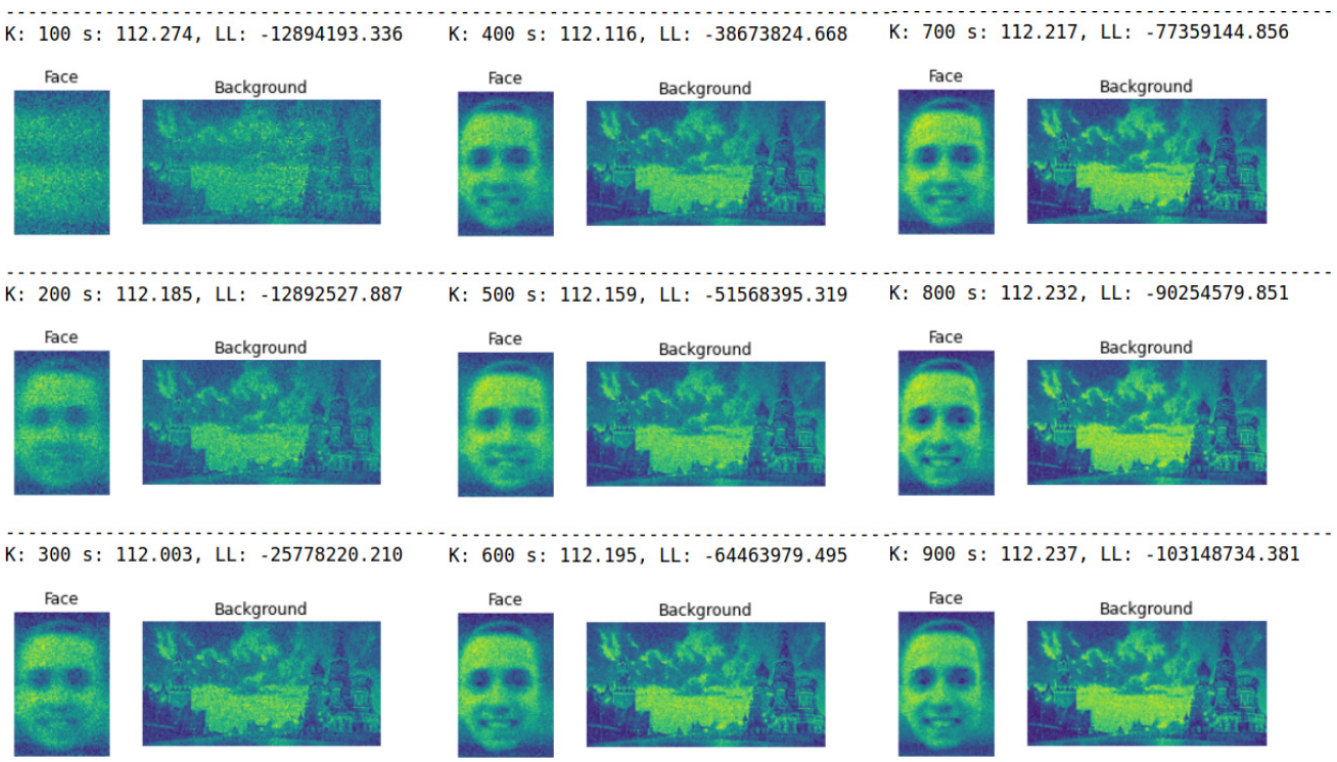
```
EM time: 2.1659, L: -88035.29997990622
MAP-EM time: 0.5916, L: -107630.98260046606
```



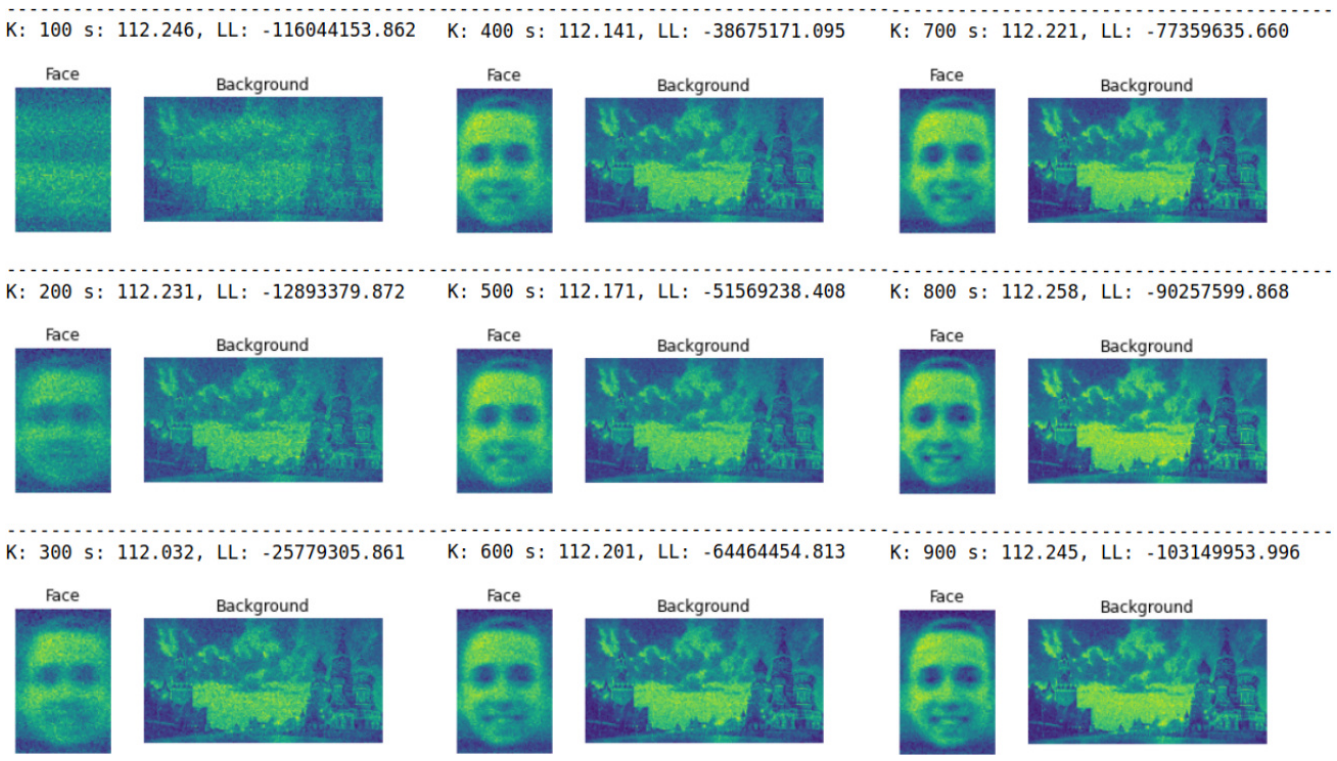
Безусловно, упрощенный EM работает быстрее - там нет трудоемких сверток. Однако по качеству он уступает классическому алгоритму, ведь при неудачном начальном приближении, когда, например, случайно инициализированное лицо совпадет где-нибудь с шумом, а значение A в этой точке случайно окажется высоким, аргмаксимум апостериорной вероятности будет определен неправильно. А поскольку MAP-EM обнуляет все остальные значения, то никаких шансов у правильной точки не будет, в то время как для классичсекого алгоритма эта точка еще могла бы найтись.
Как видно, EM восстановил лицо лучше, чем MAP-EM.

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

Классический EM:



MAP-EM:



Как видно и EM, и MAP-EM корректно работают и хорошо восстанавливают лицо и фон. Хотя во втором случае (MAP-EM) функционал качества чуть меньше, чем для EM.