Московский государственный университет имени М. В. Ломоносова

Факультет Вычислительной Математики и Кибернетики
Кафедра Математических Методов Прогнозирования
Байесовские методы машинного обучения

Отчет по практическому заданию №3

Процессы Дирихле для кластеризации изображений цифр

Выполнил:
студент 417 группы
Драпак Степан Николаевич

Москва, 2017

1 Постановка задачи

1.1 Описание модели

Рассмотрим вероятностную модель смеси распределений с априорным процессом Дирихле:

$$G \sim DP(\alpha, H), \tag{1}$$

$$\hat{\theta}_1, ...\hat{\theta}_n \sim G \tag{2}$$

$$x_n \sim p(x|\hat{\theta}_n), n = 1, N \tag{3}$$

Здесь $\alpha > 0$ – параметр концентрации, H – базовая вероятностная мера, G – вероятностная атомическая мера, x_n – наблюдаемые данные, $\hat{\theta}_n$ n – параметры компоненты смеси для объекта x_n. В силу атомичности меры G некоторые компоненты $\hat{\theta}_n$ совпадают между собой, формируя таким образом кластеры данных. Для удобства байесовского вывода модель можно представить в эквивалентном виде с помощью процесса stick-breaking:

$$p_G(\theta) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}(\theta) \tag{4}$$

$$\theta_k \sim p_H(\theta), \pi_k = v_k \prod_{i=1}^{k-1}(1 - v_i), v_k \sim Beta(v|1, \alpha) z_1, ..., z_n \sim Discret(\pi) x_n \sim p(x|\theta_{z_n}) \tag{5}$$

Здесь $\delta(\Delta)$ – дельта-функция, θ_k – атомы меры G, z_n – номер компоненты смеси для объекта x_n.
Совместное распределение для stick-breaking модели можно записать:

$$p(X, Z, v, \theta|\alpha, H) = [\prod_{k=1}^{\infty} p_H \theta_k Beta(v_k|1, \alpha)] \prod_{n=1}^{N} \prod_{k=1}^{\infty} (p(x_n|\theta_k) v_k \prod_{i=1}^{k-1}(1 - v_i))^{[z_n=k]} \tag{6}$$

Здесь бесконечные суммы можно заменить конечными, выбрав константу T, ограничивающую суммирование. Это означает, что мы делаем предположение, что в наших данных реальных кластеров будет не более чем T.

1.2 Данные

В качестве данных возьмем выборку Digits, состоящую из небольшого числа картинок размера 8x8. Вытянем все картинки в вектор и переведем их в бинарное представления, путем отсечения яркости по порогу. $\hat{x}_i = (x_i > 8)$.

В качестве одной компоненты смеси рассмотрим независимые распределения Бернулли:

$$p(x|\theta) = \prod_{i=1}^{D} \theta_i^{x_i} (1 - \theta_i)^{1 - x_i} \tag{7}$$

Здесь $x_i \in \{0, 1\}$ – i-ый пиксел изображения, $\theta_i \in (0, 1)$ – параметры компоненты. Из соображений сопряжённости в качестве априорного распределения для θ возьмём независимое Бета-распределение с общими параметрами a, b > 0:

$$p(x|\theta) = \prod_{i=1}^{D} Beta(\theta_i|a, b) \tag{8}$$

1.3 Формулировка задания

Для модели (6) с компонентами смеси (7) и априорными распределениями (8) с помощью алгоритма вариационного вывода требуется найти факторизованное приближение для апостериорного распределения: $q(Z)q(\theta)q(v) \simeq p(Z, \theta, v|X, \alpha, a, b)$.

2 Вывод формул для пересчета компонент

2.1 $q(v)$

$$\log q(v) = \mathbb{E}_{q\setminus q(v)} \log p(X, Z, v, \theta|\alpha, H) + C = \mathbb{E}_{q(\theta),q(Z)} \Bigg[[\textstyle\sum_{k=1}^{\infty} \log p_H \theta_k + \log Beta(v_k|1, \alpha)] +$$

$$+ \textstyle\sum_{n=1}^{N} \sum_{k=1}^{\infty} [z_n = k] \Big(\log p(x_n|\theta_k) + \log v_k + \sum_{i=1}^{k-1} \log(1 - v_i) \Big) \Bigg] + C$$

Выбросим независящие от θ слагаемые и внесем матожидания под суммы, где это возможно. Кроме того, константны из Beta распределения занесем в общую константу.
$\log q(v) = \sum_{k=1}^{\infty} (\alpha - 1) \log(1 - v_k) + \sum_{n=1}^{N} \sum_{k=1}^{\infty} \mathbb{E}_{q(Z)}[z_n = k] \Big(\log v_k + \sum_{i=1}^{k-1} \log(1 - v_i) \Big) + C =$
$\sum_{k=1}^{\infty} \Bigg[\log v_k \big(\sum_{n=1}^{N} \mathbb{E}_{q(Z)}[z_n = k] \big) + \log(1 - v_k) \big((\alpha - 1) + \sum_{i=k+1}^{\infty} \sum_{n=1}^{N} \mathbb{E}_{q(Z)}[z_n = i] \big) \Bigg] + C$

Тогда:

$$q(v) = \prod_{k=1}^{\infty} q(v_k) = \prod_{k=1}^{\infty} Beta(v_k|\nu_k, \mu_k)$$

Где параметры у $Beta$ распределения:
$\nu_k = a + \sum_n \mathbb{E}_{q(Z)}[z_n = k]$
$\mu_k = \alpha + \sum_{i=k+1}^{\infty} \sum_n \mathbb{E}_{q(Z)}[z_n = i]$

2.2 $q(\theta)$

Действуем аналогично. $\log q(\theta) = \mathbb{E}_{q\setminus q(\theta)} \log p(X, Z, v, \theta|\alpha, H) + C = \mathbb{E}_{q(v),q(Z)} \Bigg[\sum_{k=1}^{\infty} \log p_H \theta_k +$

$$+ \textstyle\sum_{n=1}^{N} \sum_{k=1}^{\infty} \mathbb{E}_{q(Z)}[z_n = k] \log p(x_n|\theta_k) \Bigg] + C =$$

$$\textstyle\sum_{k=1}^{\infty} \sum_{i=1}^{D} \Bigg[\log \theta_{ki} \Big[(a-1) + \sum_n \mathbb{E}_{q(Z)}[z_n = k] x_{ni} \Big] + \log(1 - \theta_{ki}) \Big[(b-1) + \sum_n \mathbb{E}_{q(Z)}[z_n = k](1 - x_{ni}) \Big] \Bigg] + C$$

Отсюда:

$$q(\theta) = \prod_{k=1}^{\infty} \prod_{i=1}^{D} q(\theta_{ki}) = \prod_{k=1}^{\infty} \prod_{i=1}^{D} Beta(\theta_{ki}|\xi_{ki}, \zeta_{ki})$$

Где параметры $Beta$ распределения:
$\xi_{ki} = a + \sum_n \mathbb{E}_{q(Z)}[z_n = k] x_{ni}$
$\zeta_{ki} = b + \sum_n \mathbb{E}_{q(Z)}[z_n = k](1 - x_{ni})$

2.3 $q(Z)$

$$\log q(Z) = \mathbb{E}_{q\setminus q(Z)} \log p(X, Z, v, \theta|\alpha, H) + C = \mathbb{E}_{q(v),q(\theta)} \Bigg[[\sum_{k=1}^{\infty} \log p_H \theta_k + \log Beta(v_k|1, \alpha)] +$$

$$+ \sum_{n=1}^{N} \sum_{k=1}^{\infty} [z_n = k] \Big(\log p(x_n|\theta_k) + \log v_k + \sum_{i=1}^{k-1} \log(1 - v_i) \Big) \Bigg] + C$$

Первое слагаемое от Z не зависит. Кроме того, исходя из того, что $p(x_n|\theta_k)$ – распределение Бернули, имеем:
$\mathbb{E}_{q(\theta)} \log p(x_n|\theta_k) = \sum_{i=1}^{D} x_{ni} \mathbb{E}_{q(\theta)} \log \theta_{ki} + (1 - x_{ni}) \mathbb{E}_{q(\theta)} \log(1 - \theta_{ki})$

Учитывая это, обозначим то, что стоит под математическим ожиданием за $\hat{\gamma}_{nk}$:
$\log \hat{\gamma}_{nk} = \sum_{i=1}^{D} x_{ni} \mathbb{E}_{q(\theta)} \log \theta_{ki} + (1 - x_{ni}) \mathbb{E}_{q(\theta)} \log(1 - \theta_{ki}) + \mathbb{E}_{q(v)} \log v_k + \sum_{i=1}^{k-1} \mathbb{E}_{q(v)} \log(1 - v_i)$

Тогда, введем γ_{nk}:

$$\gamma_{nk} = \frac{\gamma_{nk}}{\sum_i \gamma_{ni}}$$

В этих обозначениях получаем:

$$q(Z) = \prod_{n=1}^{N} \prod_{k=1}^{\infty} \gamma_{nk}^{[z_n=k]}$$

Заметим, что γ_{nk} имеет смысл вероятности принадлежности объекта с номером n к компоненте с номером k. Осталось вычислить математические ожидания в правой части формулы для $\log \hat{\gamma}_{nk}$. Для этого вспомним, что если $x \sim Beta(\alpha, \beta)$, тогда верно:

$$\mathbb{E} \log(x) = \psi(\alpha) - \psi(\alpha + \beta)$$

$$\mathbb{E} \log(1 - x) = \psi(\beta) - \psi(\alpha + \beta)$$

Тогда:

$$\mathbb{E} \log(\theta_{ki}) = \psi(\xi_{ki}) - \psi(\xi_{ki} + \zeta_{ki})$$

$$\mathbb{E} \log(1 - \theta_{ki}) = \psi(\zeta_{ki}) - \psi(\xi_{ki} + \zeta_{ki})$$

$$\mathbb{E} \log(v_k) = \psi(\nu_k) - \psi(\nu_k + \mu_k)$$

$$\mathbb{E} \log(1 - v_k) = \psi(\mu_k) - \psi(\nu_k + \mu_k)$$

Поскольку мы знаем параметры для соответствующих распределений θ и v, мы можем вычислить матрицу γ.

3 Вычисление вариационной нижней оценки

$$\mathcal{L}(q) = \mathbb{E}_q \log \Big(\frac{p(X, Z, \theta, v|\alpha, H)}{q(Z)q(\theta)q(v)} \Big)$$

Отдельно посмотрим на то, что получается от числителя и знаменателя

$$\mathcal{L}(q)_{num} = \mathbb{E}_q \Bigg[[\sum_{k=1}^{\infty} \log p_H \theta_k + \log Beta(v_k|1,\alpha)] +$$
$$\sum_{n=1}^{N} \sum_{k=1}^{\infty} [z_n = k] \Big(\log p(x_n|\theta_k) + \log v_k + \sum_{i=1}^{k-1} \log(1-v_i) \Big) \Bigg] =$$
$$\sum_{k=1}^{\infty} \Big[\sum_{i=1}^{D} \Big((a-1)\mathbb{E}_{q(\theta)} \log \theta_{ki} + (b-1)\mathbb{E}_{q(\theta)} + \log \tfrac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \Big) + (\alpha-1)\mathbb{E}_{q(v)} \log(1-v_k) + \log \tfrac{\Gamma(\alpha+1)}{\Gamma(\alpha)\Gamma(1)} \Big] +$$
$$\sum_{n=1}^{N} \sum_{k=1}^{\infty} \mathbb{E}_{q(z)}[z_n = k] \Big[\sum_{i=1}^{D} \big(x_{ni} \mathbb{E}_{q(\theta)} \log \theta_{ki} + (1-x_{ni}) \mathbb{E}_{q(\theta)} \log(1-\theta_{ki}) \big) +$$
$$+ \mathbb{E}_{q(v)} \log v_k + \sum_{i=1}^{k-1} \mathbb{E}_{q(v)} \log(1-v_i) \Big]$$

$$\mathcal{L}(q)_{denum} = \sum_{n=1}^{N} \sum_{k=1}^{\infty} \mathbb{E}_{q(Z)}[z_n = k] \log \gamma_{nk} + \sum_{k=1}^{\infty} \sum_{i=1}^{D} \mathbb{E}_{q(Z)} \log Beta(\theta_{ki}|\xi_{ki}, \zeta_{ki}) +$$
$$\sum_{k=1}^{\infty} \mathbb{E}_{q(v)} \log Beta(v_k|\nu_k, \mu_k) = \sum_{n=1}^{N} \sum_{k=1}^{\infty} \mathbb{E}_{q(Z)}[z_n = k] \log \gamma_{nk} +$$
$$\sum_{k=1}^{\infty} \sum_{i=1}^{D} (\xi_{ki} - 1)\mathbb{E}_{q(\theta)} \log \theta_{ki} + (\zeta_{ki} - 1)\mathbb{E}_{q(\theta)} \log(1-\theta_{ki}) + \log \tfrac{\Gamma(\xi_{ki}+\zeta_{ki})}{\Gamma(\xi_{ki})\Gamma(\zeta_{ki})} +$$
$$\sum_{k=1}^{\infty} (\nu_{ki} - 1)\mathbb{E}_{q(v)} \log v_k + (\zeta_{ki} - 1)\mathbb{E}_{q(v)} \log(1-v_k) + \log \tfrac{\Gamma(\nu_k+\mu_k)}{\Gamma(\mu_k)\Gamma(\nu_k)}$$

Тогда

$$\mathcal{L}(q) = \mathcal{L}(q)_{num} - \mathcal{L}(q)_{denum}$$

4 Эксперименты

Для экспериментов используем выборку digits, ~ 1800 картинок черно-белых цифр. Для того чтобы избежать бесконечного суммирования ограничим максимальное число компонент параметром T:

$$T = const * \alpha \log(1 + n/\alpha)$$

В данных экспериментах const = 10.

4.1 Подбор параметров, α

Начнем работать с половиной выборки digits. Для начала, рассмотрим влияние параметра α. Зафиксируем для этого параметры a, b и будем варировать α.
Посмотрим на центры кластеров на Рис. 1

Вообще говоря, не очевидно, на что влияет это. Исходя из формулы для оценки числа кластеров приходила в голову мысль, что чем больше α тем больше кластеров, однако, по крайней мере на этих данных, разницы нет.

4.2 Подбор параметров, a, b

Аналогично фиксируем α и b. Пробуем разобраться что меняется с a Рис. 2.

Видно, что при увеличении a число кластеров сокращается. При a = 1 и a = 0.1 каждой цифре соответствует по крайней мере один центр кластера. При a = 5 у нас остается всего 3 размытых кластера. Таким образом, получается, что параметр a говорит алгоритму, что выгодней, добавить новый кластер или по пробовать отнести объект к уже имеющемуся. Вообще, если взять картинку с beta распределением из википедии Рис. 3, то видно, что при малых a, b плотность больше концентрируется в 0 и в 1, а при их росте, начинает больше скапливаться в около 0.5. Таким образом, получается, что разных картинок в кластере много, центры получаются сильно размытые.

Аналогичная картина с параметром b. Рис. 4

В итоге, возьмем a = b = 0.1, alpha = 1.
На примере этих параметров, удостоверимся, что $\mathcal{L}$ монотонно неубывает: Рис. 5

Так выглядят центры кластеров на полной выборке с этими параметрами Рис. 6:

4.3 Эксперименты с классификацией

Возьмем 2 алгоритма классификации, градиентный бустинг и лог. регрессию. Разделим выборку на train и test. Для начала посмотрим что получится на исходных данных. Подберем параметры на train'е с помощью кросс-валидации. Лучший результаты на test'е получились примерно одинаковыми. Точность для бустинга 0.964, для лог. регрессии 0.948. После этого были проделаны аналогичные манипуляции с выборками, в которых в качестве признакового описания брались величины: $q(z_n = k)$. Для бустинга точность 0.893, для лог. регрессии 0.88. Матрица ошибок для бустинга, обученного на производных признаках, приведена на Рис. 7:

Понятно, что снижая размерность мы теряем данные, однако очевидно, что такая выборка по прежнему обладает достаточно информацией, которая дает возможность обучить вполне разумный классификатор. Интересно было бы посмотреть на большой выборке какие результаты бы получились в результате конкатинации исходных признаков и производных. Очевидно, что на такой маленькой выборки дисперсия ошибки столь велика, что никакого смысла проверять этого нет. По матрице ошибок едва ли можно сделать выводы о каких-то системных ошибках, разве что 2 и 8 часто путаются, однако есть подозрения, что это эта проблема характерна именно для данной выборки.

4.4 Последовательно добавление

Разделим нашу выборку на 4 части и будем последовательно добавлять их в алгоритме вывода. Рис. 8

Видно, что на второй итерации пропали или сильно видоизменились те кластеры с первой итерации, которые были меньше всего похожи на цифры(подчеркнуты красным). После третей же итерации мало кластеров пропало, но многие видоизменлись и стали больше похожи на реальные цифры, как например подчеркнутые девятки. На 4й итерации кластеры менялись слабее, алгоритм уже был близок к локальному максимуму.

5 Выводы

В ходе экспериментов стало ясно, что на таких данных как рукописные цифры алгоритм работает неплохо, кластеры интерпретируемы, а признаки, которые можно извлечь по результатам работы алгоритма достаточно информативны. Однако исследуемая выборка была до неприличия простая и понятно, что вообще говоря, на ней все что угодно отработает неплохо. Разумеется, огромным приемущество алгоритма является то, что он сам пытается определять необходимое число компонент, во многих задачах это будет важным плюсом. Так же, этот алгоритм можно использовать как классический метод снижения размерности, если нет возможности использовать вычислительно сложные методы. Было бы интересно сравнить его с классическими методами снижения размерности, но опять же на такой бедной выборке в этом нет смысла.

alpha=0.1, 31 Кластер

alpha=1, 32 Кластера

alpha=100, 33 Кластера

Рис. 1: Разные α, a=b=0.1

a=0.1, 28 Кластер

a=1, 17 Кластера

a=5, 3 Кластера

Рис. 2: Разные a, α = 1 b=0.1

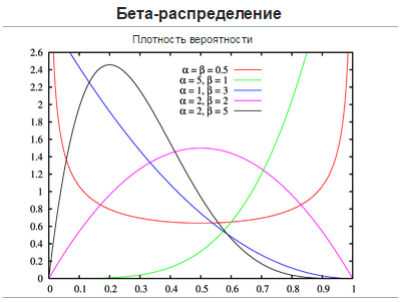

Рис. 3: Beta distribution density

b=0.1, 28 Кластер

b=1, 17 Кластера

b=5, 3 Кластера

Рис. 4: Разные b, a = 0.1, $\alpha = 1$

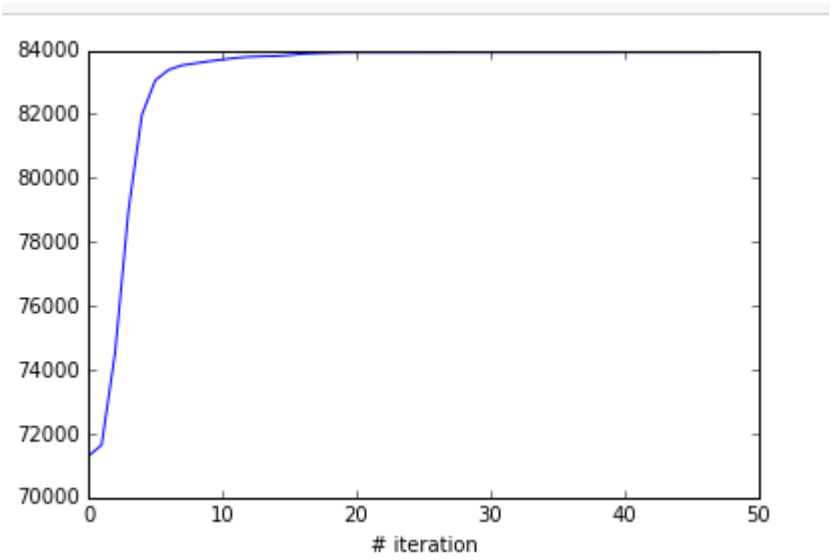

Рис. 5: $\mathcal{L}$

Рис. 6: a=b=0.1, alpha=1. Полная выборка

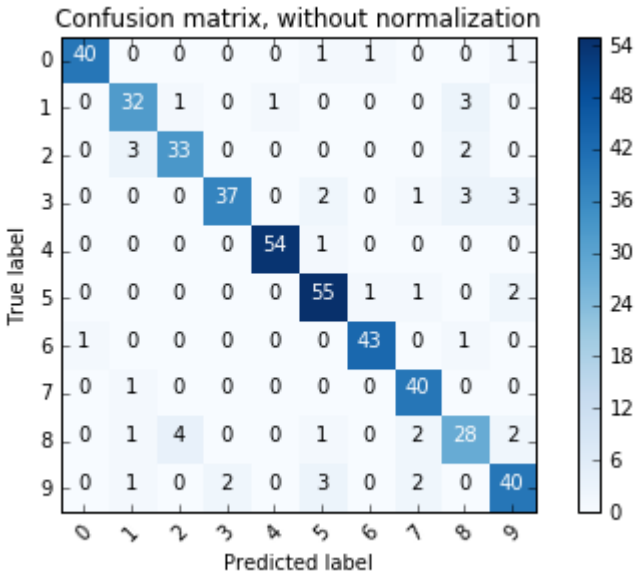

Рис. 7: Confusion matrix

Первая четверть, 31 компонента

Вторая четверть, 27 компонент

Третья четверть, 26 компонент

Четвертая четверть, 26 компонент

Рис. 8: Итеративное добавление данных