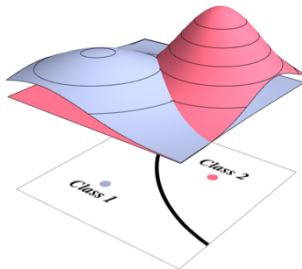


Генеративные модели

Виктор Китов

victorkitov.github.io



Курс поддержан
фондом
'Интеллект'



Победитель
конкурса VK среди
курсов по IT



Содержание

- 1 Минимизация ожидаемого штрафа и числа ошибок
- 2 Гауссов классификатор
- 3 Генеративные модели классификации текстов

Штрафы за неправильные классификации

- Предсказываем $y \in \{1, 2, \dots, C\}$
- λ_{yf} - штраф за прогноз класса y классом f .
- Примеры задач, где штрафы важны:
 - медицина: классификация болезни, методов лечения
 - финансы: детекция мошеннических сделок
 - почта: фильтрация спама
 - сети: обнаружение вторжений (intrusion detection)

Матрица штрафов

- Матрица штрафов

	$f = 1$	$f = 2$	\dots	$f = C$
$y = 1$	λ_{11}	λ_{12}	\dots	λ_{1C}
$y = 2$	λ_{21}	λ_{22}	\dots	λ_{2C}
\dots	\dots	\dots	\dots	\dots
$y = C$	λ_{C1}	λ_{C2}	\dots	λ_{CC}

- Ожидаемая цена прогноза $\hat{y}(x) = f$:

$$\mathcal{L}(f) = \sum_y p(y|x) \lambda_{yf}$$

- Байесовское правило минимального риска
 - англ. Bayes minimum risk decision rule

$$\hat{y}(x) = \arg \min_f \mathcal{L}(f)$$

Упрощение решающего правила

УПРОЩЕНИЕ 1: за *любые* ошибки на классе y платим λ_y .

$$\lambda_{yf} \equiv \lambda_y \mathbb{I}[y \neq f]$$

Упрощение решающего правила

УПРОЩЕНИЕ 1: за *любые* ошибки на классе y платим λ_y .

$$\lambda_{yf} \equiv \lambda_y \mathbb{I}[y \neq f]$$

Матрица штрафов:

	$f = 1$	$f = 2$	\dots	$f = C$
$y = 1$	0	λ_1	\dots	λ_1
$y = 2$	λ_2	0	\dots	λ_2
\dots	\dots	\dots	\dots	\dots
$y = C$	λ_C	λ_C	\dots	0

Упрощение решающего правила

- Ожидаемый штраф за прогноз f :

$$\mathcal{L}(f) = \sum_y p(y|x) \lambda_y \mathbb{I}[f \neq y] = \sum_y p(y|x) \lambda_y - p(f|x) \lambda_f$$

- Байесовское правило минимального риска становится:

$$\begin{aligned} \hat{y}(x) &= \arg \min_f \mathcal{L}(f) = \arg \min_f \left(\overbrace{\sum_y p(y|x) \lambda_y}^{\text{const}(f)} - p(f|x) \lambda_f \right) = \\ &= \arg \min_f (-p(f|x) \lambda_f) = \arg \max_f \lambda_f p(f|x) \end{aligned} \quad (1)$$

- Важна не только вероятность класса, но и штраф при пропуске класса.

Упрощение решающего правила

- **УПРОЩЕНИЕ 2:** одинаковый штраф при любых ошибках $\lambda_y \equiv \lambda \forall y$.
- Байесовское правило минимального риска становится

$$\hat{y}(x) = \arg \max_f p(f|x) \quad (2)$$

- Это Байесовское правило минимальной ошибки.
 - т.к. прогноз максимально вероятным классом минимизирует ожидаемое число ошибок.

Упрощение решающего правила

- **УПРОЩЕНИЕ 2:** одинаковый штраф при любых ошибках $\lambda_y \equiv \lambda \forall y$.
- Байесовское правило минимального риска становится

$$\hat{y}(x) = \arg \max_f p(f|x) \quad (2)$$

- Это Байесовское правило минимальной ошибки.
 - т.к. прогноз максимально вероятным классом минимизирует ожидаемое число ошибок.
- **УПРОЩЕНИЕ 3:** Если x и y независимы, то $p(f|x) = p(f)$ и (2) становится

$$\hat{y}(x) = \arg \max_f p(f|x) = \arg \max_f p(f)$$

Генеративные и дискриминативные модели

$$\hat{y}(x) = \arg \max_y p(y|x) = \arg \max_y \frac{p(x, y)}{p(x)} = \arg \max_y p(y)p(x|y)$$

Можно строить прогноз по

- $p(y|x)$: дискриминативная модель
 - моделируем только то, что нужно; простота оценивания
- $p(y)p(x|y) = p(x, y)$: генеративная модель
 - $p(y)$ легко оценить, $p(x|y)$ - сложно
 - возможное упрощение: предположение наивного Байеса

$$p(x|y) = p(x^1|y)p(x^2|y)...p(x^D|y)$$

- можно подстраивать модель под изменяемые $p(y)$
- если x^i пропущено, то оценивается

$$p(y) p(x \setminus \{x^i\} | y) = p(y) \int_{x^i} p(x|y) dx^i$$

- легко фильтровать выбросы - малое $p(x)$

Содержание

- 1 Минимизация ожидаемого штрафа и числа ошибок
- 2 Гауссов классификатор**
- 3 Генеративные модели классификации текстов

Гауссов классификатор

- Гауссов классификатор - генеративная модель с $x|y \sim \mathcal{N}(\mu_y, \sigma_y^2)$:

$$p(x|y) = \frac{1}{(2\pi)^{D/2} |\Sigma_y|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) \right\}$$

Гауссов классификатор

- Гауссов классификатор - генеративная модель с $x|y \sim \mathcal{N}(\mu_y, \Sigma_y)$:

$$p(x|y) = \frac{1}{(2\pi)^{D/2} |\Sigma_y|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) \right\}$$

- Дискриминантная функция

$$\begin{aligned} \log p(y|x) &= \log p(x|y) + \log p(y) - \log p(x) \\ &= -\frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) - \frac{1}{2} \log |\Sigma_y| \\ &\quad - \frac{D}{2} \log(2\pi) + \log p(y) - \log p(x) \end{aligned}$$

Гауссов классификатор

- Гауссов классификатор - генеративная модель с $x|y \sim \mathcal{N}(\mu_y, \sigma_y^2)$:

$$p(x|y) = \frac{1}{(2\pi)^{D/2} |\Sigma_y|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) \right\}$$

- Дискриминатная функция

$$\begin{aligned} \log p(y|x) &= \log p(x|y) + \log p(y) - \log p(x) \\ &= -\frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) - \frac{1}{2} \log |\Sigma_y| \\ &\quad - \frac{D}{2} \log(2\pi) + \log p(y) - \log p(x) \end{aligned}$$

- Уберем общие для всех дискр. ф-ций константы:

$$g_y(x) = \log p(y) - \frac{1}{2} \log |\Sigma_y| - \frac{1}{2} (x - \mu_y)^T \Sigma_y^{-1} (x - \mu_y) \quad (3)$$

Практическое применение

- Заменяем $p(y)$, μ_y , Σ_y их оценками макс. правдоподобия:

$$\hat{p}(y) = \frac{N_y}{N}, \quad \hat{\mu}_y = \frac{1}{N_y} \sum_{n:y_n=y} x_n$$

$$\hat{\Sigma}_y = \frac{1}{N_y} \sum_{n:y_n=y} (x_n - \hat{\mu}_y)(x_n - \hat{\mu}_y)^T$$

- Модель опирается на предположение $x|y \sim \mathcal{N}(\mu_y, \sigma_y^2)$, в частности, унимодальность распределения.
- $p(y)$: 1 параметр, μ_y : D параметров
- Σ_y : $\frac{D(D+1)}{2}$ параметров
- Всего параметров для $y = 1, 2, \dots, C$:

$$C \left(1 + D + \frac{D(D+1)}{2} \right)$$

Упрощение модели

- Гауссов классификатор квадратично зависит от #признаков.
- Упрощающие предположения:
 - $\Sigma_1, \Sigma_2, \dots, \Sigma_C$ - диагональные (naïve Bayes)
 - уменьшить #признаков (отбор признаков / снижение размерности)
 - $\Sigma_1 = \Sigma_2 = \dots = \Sigma_C = \Sigma$
 - $\Sigma_1 = \alpha_1 \Sigma, \Sigma_2 = \alpha_2 \Sigma, \dots, \Sigma_C = \alpha_C \Sigma$.

Регуляризация модели

- Если число наблюдений класса y мало, а D велико, то Σ_y может получиться вырожденной.
- Регуляризация для обратимости и плавного контроля сложности:

$$\begin{aligned}\Sigma'_y &= \Sigma_y + \lambda I \\ \Sigma'_y &= \Sigma_y + \lambda \text{diag}\{\Sigma_y\} \\ \lambda &> 0\end{aligned}$$

QDA vs. LDA

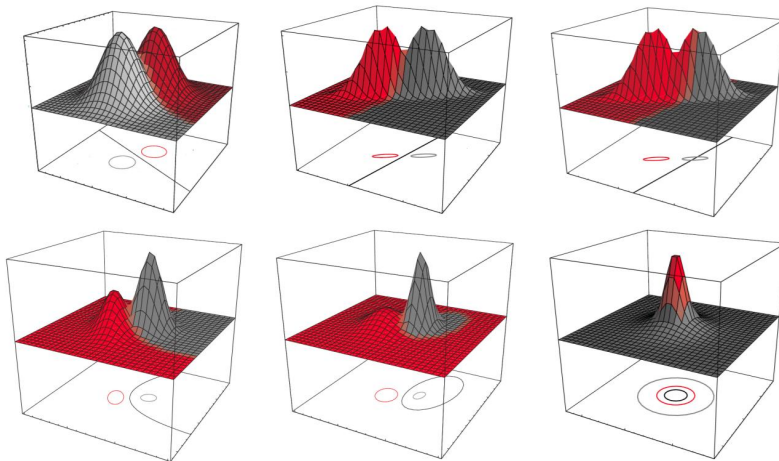
Метод Гауссова классификатора называется:

- *квадратичным дискриминантным анализом*, когда $\Sigma_1, \Sigma_2, \dots, \Sigma_C$ - произвольные.
 - границы между классами квадратичные¹
- *линейным дискриминантным анализом*, когда $\Sigma_1 = \Sigma_2 = \dots = \Sigma_C$ общие
 - др. название - линейный дискриминант Фишера
 - границы между классами линейные²

¹ Докажите.

² Докажите.

Линейный и квадратичный дискриминант



LDA (вверху) и QDA (внизу): $p(x|y)$, границы.

Содержание

- 1 Минимизация ожидаемого штрафа и числа ошибок
- 2 Гауссов классификатор
- 3 Генеративные модели классификации текстов

Токены в текстах

Требуется представить текст вектором $\in \mathbb{R}^D$

Будем учитывать встречаемость D токенов w_1, w_2, \dots, w_D

- в простейшем случае: все уникальные слова языка
 - можно в разных формах или нормализованной
 - единственное число, именительный падеж, начальная форма глагола.
- убрать слишком частые слова и слишком редкие
- убрать неинформативные "стоп-слова" из словаря
 - а, но, если, конечно, зато, или, ...

Токены в текстах

- можно ограничить словами предметной области
- можно добавить биграммы/триграммы:
 - мне фильм не понравился -> 'мне фильм', 'фильм не', 'не понравился'.
 - мне фильм не понравился -> 'мне фильм не', 'фильм не понравился'.
- либо можно добавить только коллокации (неслучайно часто встречающиеся слова)
 - линейная регрессия показала точность... -> 'линейная регрессия'

$$\frac{p(w_1 w_2)}{p(w_1)p(w_2)} > threshold$$

Генеративные модели классификации текстов

- Генеративные модели классификации текстов:
 - Модель Бернулли
 - $x^i = \mathbb{I}[w_i \text{ встретилось в документе}]$
 - Мультиномиальная
 - $x^i = [\text{сколько раз } w_i \text{ встретилось в документе}]$
- Могут применяться и в др. областях:
 - изображения (кодирование bag-of-visual-words)
 - ДНК - цепочка нуклеотидов
 - покупки в магазине
 - использованные услуги в тарифе

Модель Бернулли³

- w_1, w_2, \dots, w_D - токены
- $x \in \mathbb{R}^D$, $x^d = \mathbb{I}[w_d \text{ встретилось в документе}]$, $d = \overline{1, D}$
- $N = \#[\text{документов}]$, $N^y := \#[\text{документов класса } y]$
- $N_d^y = \#[\text{документов класса } y, \text{ содержащих } d\text{-й токен}]$
- $\theta_d^y = p(x^d = 1|y)$
- Частотные оценки (макс. правдоподобия):

$$p(y) \approx \frac{N^y}{N}, \quad \theta_d^y \approx \frac{N_d^y}{N^y}$$

³Является ли она линейным классификатором?

Модель Бернулли⁶

- Решающее правило (минимальной ошибки):

$$\hat{y}(x) = \arg \max_y p(y)p(x|y)$$

- Генерация документа класса y : для каждого токена w_d генерируется его присутствие в документе $\sim \text{Bernoulli}(\theta_y^d)$.

- не зависит от встречаемости др. токенов (naive Bayes)

- $p(x|y) = \prod_{d=1}^D (\theta_y^d)^{x^d} (1 - \theta_y^d)^{1-x^d}$

- Сглаживание Лапласа^{4,5}: $\theta_y^d = \frac{N_d^y + \alpha}{N^y + 2\alpha}$

⁴Проинтерпретируйте добавлением новых наблюдений в выборку.

⁵Как сглаживать, чтобы приближать к априорному распределению слов?

⁶Оцените сложность обучения модели Бернулли.

Мультиномиальная модель

- w_1, w_2, \dots, w_D - токены
- Решающее правило (минимальной ошибки):

$$\hat{y}(x) = \arg \max_y p(y)p(x|y)$$

- $x \in \mathbb{R}^D$, x^i = [сколько раз w_i встретилось в документе].
- Генерация документа класса y : для каждой словопозиции $i = 1, 2, \dots, n_{document}$ сгенерировать токен $z_i \sim \text{Categorical}(\theta_1^y, \theta_2^y, \dots, \theta_D^y)$.
- θ_i^y = [вероятность w_i на словопозиции]
 - не зависит от встречаемости др. токенов (naive Bayes)

Мультиномиальная модель⁷

$$p(x|y) = \frac{(\sum_i x^i)!}{\prod_i (x^i)!} \prod_{i=1}^D (\theta_i^y)^{x^i} \quad \text{мультиномиальное распределение}$$

Интерпретация мультиномиального коэффициента:

- $(\sum_i x^i)! = n!$ - число перестановок различных токенов
- перестановки токена 1 неразличимы \Rightarrow делим на $(x^1)!$
- перестановки токена 2 неразличимы \Rightarrow делим на $(x^2)!$
- ... в итоге: $\frac{(\sum_i x^i)!}{\prod_i (x^i)!}$ - #способов расставить w_1, \dots, w_D в количествах x^1, \dots, x^D по $n = \sum_i x^i$ ячейкам.

⁷Является ли она линейным классификатором?

Вывод мультиномиального коэффициента

#число способов, как можно расставить D слов по n позициям в количествах x_1, \dots, x_D :

$$\begin{aligned}
 & C_{x_1}^n C_{x_2}^{n-x_1} \dots C_{x_{D-1}}^{n-x_1-\dots-x_{D-2}} C_{x_D}^{n-x_1-\dots-x_{D-1}} \\
 &= \frac{n!}{x_1!(n-x_1)!} \times \frac{(n-x_1)!}{x_2!(n-x_1-x_2)!} \times \\
 &\dots \times \frac{(n-x_1-x_{D-2})!}{x_{D-1}!(n-x_1-\dots-x_{D-1})!} \frac{(n-x_1-\dots-x_{D-1})!}{x_D!(n-x_1-\dots-x_D)!} \\
 &= \frac{(\sum_i x^i)!}{\prod_i (x^i)!}
 \end{aligned}$$

Оценки параметров¹⁰

- Частотные оценки (макс. правдоподобия):
 - $\hat{p}(y) = \frac{N^y}{N}$, где
 - $N = \#[\text{документов}]$
 - $N^y = \#[\text{документов} \in y]$,
 - $\theta_i^y \approx n_i^y / n^y$, где
 - $n_i^y = \#[\text{токен } w_i \text{ встречался в документах} \in y]$,
 - $n^y = \#[\text{токенов в документе} \in y]$,
- Сглаживание Лапласа^{8,9}:

$$\theta_y^d = \frac{n_{yd} + \alpha}{n_y + \alpha D}$$

⁸Проинтерпретируйте добавлением новых наблюдений в выборку.

⁹Как сглаживать, чтобы приближать к априорному распределению слов?

¹⁰Оцените сложность обучения мультиномиальной модели.

Заключение

- Байесовское правило
 - минимального риска: минимизирует ожидаемые потери
 - минимальной ошибки: минимизирует #ошибок
 - оптимально в случае одинакового штрафа для любых ошибок
- Генеративные модели моделируют $p(y)p(x|y) = p(x, y)$.
- Предположение наивного Байеса:

$$p(x|y) = p(x^1|y)p(x^2|y)...p(x^D|y)$$

- Дискриминативные модели моделируют только $p(y|x)$.
 - предпочтительны в большинстве случаев
- Модели Бернулли и мультиномиальная имеют линейную относительно длин текстов сложность оценивания.
 - естественный бейзлайн для классификации текстов.