

Разбор летучки

Лекция 5

Байесовские методы классификации

Екатерина Тузова



Мотивирующий пример

Классификация сообщений

Датасет **20 newsgroups** содержит почти 20000 сообщений из списков рассылки Usenet.

Классификация сообщений

Датасет **20 newsgroups** содержит почти 20000 сообщений из списков рассылки Usenet.

Примеры сообщений из списка рассылки sci.crypt:

*When you find out a floppy password protect program,
could you e-mail me.*

Thanks

*Not to mention Computer Associates. I'll have to be careful
to stop telling people I'm a Clipper programmer, they might
lynch me... :-)*

Классификация сообщений

Датасет **20 newsgroups** содержит почти 20000 сообщений из списков рассылки Usenet.

Примеры сообщений из списка рассылки sci.crypt:

*When you find out a floppy password protect program,
could you e-mail me.*

Thanks

*Not to mention Computer Associates. I'll have to be careful
to stop telling people I'm a Clipper programmer, they might
lynch me... :-)*

Задача: Построить классификатор, предсказывающий по тексту сообщения список рассылки, в который оно было отправлено.

Вероятностная постановка задачи

X – множество объектов

Y – множество меток классов

$X \times Y$ – вероятностное пространство с плотностью $p(x, y)$

Вероятностная постановка задачи

X – множество объектов

Y – множество меток классов

$X \times Y$ – вероятностное пространство с плотностью $p(x, y)$

$$p(x, y) = P_y p(x|y)$$

P_y – априорная вероятность класса y

$p(x|y)$ – функция правдоподобия класса y

Вероятностная постановка задачи

X – множество объектов

Y – множество меток классов

$X \times Y$ – вероятностное пространство с плотностью $p(x, y)$

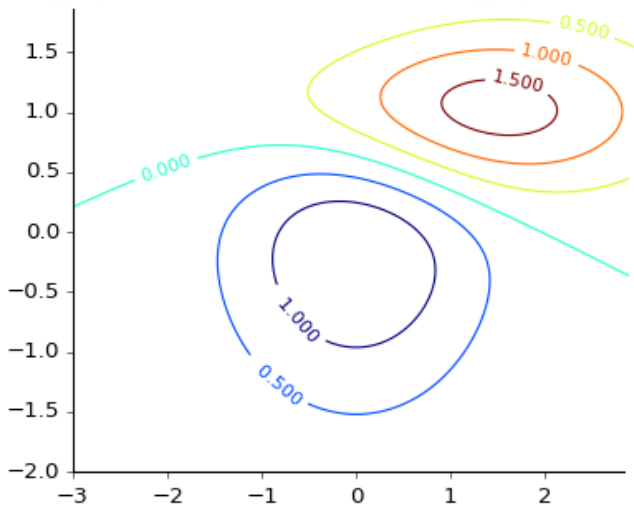
$$p(x, y) = P_y p(x|y)$$

P_y – априорная вероятность класса y

$p(x|y)$ – функция правдоподобия класса y

Задача: Построить алгоритм $a : X \rightarrow Y$, минимизирующий **вероятность** ошибки.

Плотности $p(x|y)$



Как правило, априорные вероятности P_y и функции правдоподобия классов $p(x|y)$ неизвестны.

Как правило, априорные вероятности P_y и функции правдоподобия классов $p(x|y)$ неизвестны.

2 подзадачи:

1. По выборке X^l из неизвестного распределения с плотностью $p(x, y)$ построить оценки вероятностей \hat{P}_y и функций правдоподобия $\hat{p}(x|y)$ для каждого класса
2. По известным P_y и $p(x|y)$ построить функцию $a(x)$, минимизирующую вероятность ошибочной классификации

Предположим, что нам известно заранее распределение с плотностью $p(x, y)$. Как оценить вероятность ошибочной классификации для произвольного алгоритма $a : X \rightarrow Y$?

$a : X \rightarrow Y$ разбивает X на непересекающиеся области:

$$A_s = \{x \in X | a(x) = s\} \quad s \in Y$$

$a : X \rightarrow Y$ разбивает X на непересекающиеся области:

$$A_s = \{x \in X | a(x) = s\} \quad s \in Y$$

Ошибка: объект x класса y попадает в A_s , $s \neq y$

Вероятность ошибки: $p(A_s, y) = \int_{A_s} p(x, y) dx$

Идея: Введем λ_y – штраф за назначение неправильного класса объекту из y

Идея: Введем λ_y – штраф за назначение неправильного класса объекту из y

Функционал среднего риска алгоритма a :

$$R(a) = \sum_{s \in Y} \sum_{y \in Y} \lambda_y P_y p(A_s | y)$$

Оптимальный Байесовский классификатор

Минимум среднего риска $R(a)$ достигается при:

$$a(x) = \arg \max_{y \in Y} \lambda_y P_y p(x|y)$$

Принцип максимума апостериорной вероятности

Апостериорная вероятность класса y для объекта x :

$$P(y|x) = \frac{p(x, y)}{p(x)} = \frac{P_y p(x|y)}{\sum_{s \in Y} P_s p(x|s)} \propto P_y p(x|y)$$

Принцип максимума апостериорной вероятности

Апостериорная вероятность класса y для объекта x :

$$P(y|x) = \frac{p(x, y)}{p(x)} = \frac{P_y p(x|y)}{\sum_{s \in Y} P_s p(x|s)} \propto P_y p(x|y)$$

Перепишем оптимальный алгоритм с использованием апостериорных вероятностей:

$$a(x) = \arg \max_{y \in Y} \lambda_y P(y|x)$$

Если $\lambda_y = 1$, то алгоритм максимизирует апостериорную вероятность для объекта x .

Чего еще не хватает для **работающего** классификатора?

1. Что представляют из себя X и Y для сообщений из списков рассылки

На полпути к классификатору

1. Что представляют из себя X и Y для сообщений из списков рассылки
2. Выбрать функции правдоподобия $p(x|y)$

На полпути к классификатору

1. Что представляют из себя X и Y для сообщений из списков рассылки
2. Выбрать функции правдоподобия $p(x|y)$
3. Научиться оценивать априорные вероятности классов \hat{P}_y и функции правдоподобия $\hat{p}(x|y)$ из данных

$V = v_1, \dots, v_{|V|}$ – упорядоченное множество слов

Сообщение можно представить в виде вектора, в котором на j -ой позиции стоит 1, если v_d встречается в сообщении, и 0 в противном случае

$X \equiv \{0, 1\}^{|V|}$, а Y это множество идентификаторов рассылки

V = who, I, let, dogs, out, the

$V = \text{who, I, let, dogs, out, the}$

Сообщение «Who let the dogs out? Who, who, who, who?» будет векторизовано как $[1, 0, 1, 1, 1, 1]$.

$V =$ who, I, let, dogs, out, the

Сообщение «Who let the dogs out? Who, who, who, who?» будет векторизовано как $[1, 0, 1, 1, 1, 1]$.

Как будет векторизовано предложение «Well, if I am a dog, the party is on [...]»?

Как определить функцию правдоподобия для сообщения, представленного в виде бинарного вектора?

Наивность

Идея: будем использовать дискретное распределение на множестве X , то есть сопоставим вероятность θ_{yx} каждому значению $x \in X$, тогда

$$p(x|y) = \theta_{yx}$$

Функция правдоподобия

Идея: Предположим, что все признаки (компоненты вектора x) независимы **при условии** y , тогда:

$$p(x|y) = \prod_{d=1}^{|V|} p(x^d|y)$$

Функция правдоподобия

Идея: Предположим, что все признаки (компоненты вектора x) независимы **при условии** y , тогда:

$$p(x|y) = \prod_{d=1}^{|V|} p(x^d|y)$$

Полученный классификатор называют **наивным** Байесовским классификатором из-за наивности сделанного предположения

$$a(x) = \arg \max_{y \in Y} \lambda_y P_y \prod_{d=1}^{|V|} p(x^d|y)$$

Распределение Бернулли – дискретное распределение на множестве $0, 1$ с параметром $\theta \in [0, 1]$ — вероятностью **успеха** и функцией вероятности:

$$Ber(x; \theta) = \theta^x (1 - \theta)^{1-x}$$

Распределение Бернулли

Распределение Бернулли – дискретное распределение на множестве $0, 1$ с параметром $\theta \in [0, 1]$ — вероятностью **успеха** и функцией вероятности:

$$\text{Ber}(x; \theta) = \theta^x (1 - \theta)^{1-x}$$

$$p(x|y) = \prod_{d=1}^{|V|} \theta_{yd}^x (1 - \theta_{yd})^{1-x}$$

Попробуем оценить параметры функций правдоподобия по выборке X^l .

$$L(\theta) = \sum_{i=1}^l \ln p(x_i; \theta) \rightarrow \max_{\theta}$$

Как записать условие оптимума?

$$\frac{\partial}{\partial \theta} L(\theta) = \sum_{i=1}^l \frac{\partial}{\partial \theta} \ln p(x_i, \theta) = \sum_{i=1}^l \frac{x_i}{\theta} + \frac{1 - x_i}{\theta - 1} = 0$$

$$\begin{aligned}\frac{\partial}{\partial \theta} L(\theta) &= \sum_{i=1}^l \frac{\partial}{\partial \theta} \ln p(x_i, \theta) = \sum_{i=1}^l \frac{x_i}{\theta} + \frac{1-x_i}{\theta-1} = 0 \\ \Rightarrow \hat{\theta}_{ML} &= \frac{1}{l} \sum_{i=1}^l x_i\end{aligned}$$

Наивный Байесовски классификатор

Оценим методом максимального правдоподобия априорные вероятности классов \hat{P}_y и параметры распределения Бернулли $\hat{\theta}_{yd}$

$$\hat{P}_y = \frac{\sum_{i=1}^l [y_i = y]}{l} \quad \hat{\theta}_{yd} = \frac{\sum_{i=1}^l [y_i = y] x_{id}}{\sum_{i=1}^l [y_i = y]}$$

Наивный Байесовски классификатор

Оценим методом максимального правдоподобия априорные вероятности классов \hat{P}_y и параметры распределения Бернулли $\hat{\theta}_{yd}$

$$\hat{P}_y = \frac{\sum_{i=1}^l [y_i = y]}{l} \quad \hat{\theta}_{yd} = \frac{\sum_{i=1}^l [y_i = y] x_{id}}{\sum_{i=1}^l [y_i = y]}$$

$$a(x) = \arg \max_{y \in Y} \lambda_y \hat{P}_y \prod_{d=1}^{|V|} \text{Ber}(x_{id}; \hat{\theta}_{yd})$$

Пример

$y \in Y$	\hat{P}_y	password	program	PGP
sci.crypt	0.4	0.8	0	1
comp.graphics	0.6	0.2	0.6	0

Пример

$y \in Y$	\hat{P}_y	password	program	PGP
sci.crypt	0.4	0.8	0	1
comp.graphics	0.6	0.2	0.6	0

Какой класс будет назначен сообщению
«How should I add PGP support to my program?»,
если $\forall y \in Y (\lambda_y = 1)$?

Пример

$y \in Y$	\hat{P}_y	password	program	PGP
sci.crypt	0.4	0.8	0	1
comp.graphics	0.6	0.2	0.6	0

Какой класс будет назначен сообщению
«How should I add PGP support to my program?»,
если $\forall y \in Y (\lambda_y = 1)$?

$$\begin{aligned} a(x) &= \arg \max_{y \in Y} \hat{P}(y|x = [0, 1, 1]) \\ &= \arg \max_{y \in Y} \{\hat{p}(\text{sci.crypt}|x), \hat{p}(\text{comp.graphics}|x)\} \\ &= \arg \max_{y \in Y} \{0, 0\} \end{aligned}$$

Идея: Введем параметр $\alpha \geq 0$ и добавим его в ОМП для распределения Бернулли.

$$\hat{\theta}_{yd}^* = \frac{\sum_{i=1}^l [y_i = y] x_{id} + \alpha}{\sum_{i=1}^l [y_i = y] + 2\alpha}$$

Если в обучающей выборке много представителей класса y , содержащих слово v_d , то $\hat{\theta}_{yd}^*$ будет стремиться к ОМП, в обратном случае $\hat{\theta}_{yd}^* \approx \frac{1}{2}$

+ Просто реализовать и использовать

- + Просто реализовать и использовать
- + Можно обучать по потоку данных

Вопросы?

Фреквентистский и Байесовский подходы

Фреквентистский подход предполагает, что параметры распределения некоторой случайной величины – это фиксированные (но, возможно, неизвестные) значения.

Фреквентистский и Байесовский подходы

Фреквентистский подход предполагает, что параметры распределения некоторой случайной величины – это фиксированные (но, возможно, неизвестные) значения.

Байесовский подход считает все величины случайными, то есть у параметров тоже есть распределение:

$$p(x|\theta) = \text{Ber}(x|\theta) \quad p(\theta) = \text{Beta}(\theta|\alpha, \beta)$$

Таким образом, при Байесовском подходе нас интересует не точечная оценка параметра $\hat{\theta}$, а его апостериорное распределение:

Фреквентистский и Байесовский подходы

Фреквентистский подход предполагает, что параметры распределения некоторой случайной величины – это фиксированные (но, возможно, неизвестные) значения.

Байесовский подход считает все величины случайными, то есть у параметров тоже есть распределение:

$$p(x|\theta) = \text{Ber}(x|\theta) \quad p(\theta) = \text{Beta}(\theta|\alpha, \beta)$$

Таким образом, при Байесовском подходе нас интересует не точечная оценка параметра $\hat{\theta}$, а его апостериорное распределение:

$$p(\theta|x) = \frac{p(\theta)p(x|\theta)}{\int p(\theta)p(x|\theta)d\theta} \propto p(\theta)p(x|\theta)$$

Сопряжённое априорное распределение

Наблюдение: Часто можно выбрать априорное распределение $p(\theta)$ таким образом, чтобы апостериорное распределение $p(\theta|x)$ имело тот же вид, что и априорное, только с другими параметрами.

Сопряжённое априорное распределение

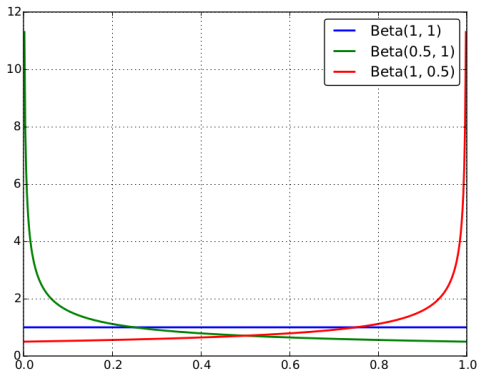
Наблюдение: Часто можно выбрать априорное распределение $p(\theta)$ таким образом, чтобы апостериорное распределение $p(\theta|x)$ имело тот же вид, что и априорное, только с другими параметрами.

Семейство распределений $p(\theta|\alpha)$ называется априорным сопряжённым для семейства функций правдоподобия $p(x|\theta)$, если апостериорное распределение $p(\theta|x, \alpha)$ остаётся в том же семействе:

$$p(\theta|x, \alpha) \propto p(\theta)p(x|\theta) = p(\theta|\alpha^*)$$

α и α^* – это гиперпараметры, то есть параметры распределения параметров.

Бета-распределение



$$p(\theta|\alpha, \beta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)} \quad B(\alpha, \beta) = \int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1} d\theta$$

И снова про аддитивное сглаживание

Бета-распределение является априорным сопряжённым для распределения Бернулли

$$\begin{aligned} p(\theta|x, \alpha, \beta) &\propto p(\theta|\alpha, \beta)p(x|\theta) \\ &\propto (\theta^{\alpha-1}(1-\theta)^{\beta-1})(\theta^x(1-\theta)^{1-x}) \\ &\propto \theta^{(\alpha+x)-1}(1-\theta)^{\beta+1-x} = \text{Beta}(\theta|\alpha^*, \beta^*) \end{aligned}$$

Оценка параметра θ

$$\begin{aligned}\hat{\theta} &= \int_0^1 \theta p(\theta|x, \alpha, \beta) d\theta = \mathbb{E}[Beta(\theta|\alpha^*, \beta^*)] \\ &= \frac{\alpha^*}{\alpha^* + \beta^*} = \frac{\alpha + x}{\alpha + \beta + 1}\end{aligned}$$

Если $\alpha = \beta$, то $\hat{\theta}$ в точности совпадает со сглаженной оценкой $\hat{\theta}^*$

Что почитать по этой лекции

- Kevin P. Murphy "Machine Learning: A Probabilistic Perspective" Chapter 3
- Воронцов "Байесовские алгоритмы классификации"
- Tom Mitchell "Machine Learning" Chapter 6

На следующей лекции

- Перцептрон
- Функция потерь
- Преоброессинг
- Ошибка обобщения