Урок 11

Байесовская классификация и регрессия

11.1. Спам-фильтр на основе байесовского классификатора

Этот раздел посвящен наивному байесовскому классификатору на примере практической задачи фильтрации спама.

11.1.1. Задача фильтрации спама

Фильтр спама представляет собой бинарный классификатор:

$$Y = \{\underset{\text{спам}}{spam}, \underset{\text{не спам}}{ham}\}.$$

Первые спам-фильтры являлись наивными байесовскими классификаторами.

Примеры спамных писем:

- Hi! :) Purchase Exclusive Tabs Online http://...
- We Offer Loan At A Very Low Rate Of 3%. If Interested, Kindly Contact Us, Reply by email ...@hotmail.com
- Купите специализацию Машинное обучение и анализ данных от МФТИ и Яндекса с супер-скидкой 0.99%! Станьте Data Scientist за 5 месяцев!

Отчетливо видно, что некоторые слова особенно часто встречаются в спаме.

Пусть есть коллекция писем, среди которых n_s писем — спам, а n_h — не спам. Возникает идея подсчитать для каждого такого слова w количество n_{ws} писем со спамом и количество n_{wh} писем без спама, в которых есть это слова. Тогда можно оценить вероятность появления каждого слова в спамном и неспамном письме:

$$P(w|spam) = \frac{n_{ws}}{n_s}, \qquad P(w|ham) = \frac{n_{wh}}{n_h}.$$

Пусть теперь требуется выяснить является ли некоторый новый текст, содержащий ключевые слова $w_1, ..., w_N$, спамом или нет. Можно оценить следующие вероятности порождения текста, если известно, что он принадлежит какому-либо из классов:

$$P(text|spam) = P(w_1|spam)P(w_2|spam)...P(w_N|spam),$$
$$P(text|ham) = P(w_1|ham)P(w_2|ham)...P(w_N|ham).$$

Такую оценку можно сделать только в случае, когда вероятность вхождения разных слов в текст — независимые события. Это достаточно наивное предположение, поэтому классификатор называется «наивный байесовский классификатор».

11.1.2. Идея наивного байесовского классификатора

В качестве алгоритма можно использовать следующий: выбрать такой класс, вероятность порождения текста в котором больше:

$$a(text) = \operatorname{argmax}_y P(text|y).$$

Это почти правильный алгоритм, но, поскольку текст уже известен, более правильно оценивать вероятность $P(y|text)$ того, что этот текст принадлежит какому-то из классов:

$$a(text) = \text{argmax}_y\, P(y|text).$$

Эту вероятность можно вычислить используя теорему Байеса:

$$P(y|text) = P(text|y)P(y)/P(text)$$

Тогда, поскольку $P(text)$ не содержит зависимость от y:

$$a(text) = \text{argmax}_y\, P(text|y)P(y)/P(text) = \text{argmax}_y\, P(text|y)P(y).$$

11.1.3. Спам фильтр на наивном байесовском классификаторе

Для того, чтобы построить спам фильтр на наивном байесовском классификаторе, на стадии обучения необходимо оценить вероятности вхождения слов в тексты каждого из классов:

$$P(w|spam) = \frac{n_{ws}}{n_s}, \qquad\qquad P(w|ham) = \frac{n_{wh}}{n_h}.$$

На стадии применения классификатора к текстам необходимо выбрать такой класс $y \in \{spam, ham\}$, для которого произведение:

$$P(y)P(text|y) = P(y)P(w_1|y)...P(w_N|y)$$

максимально. Именно к этому классу и следует отнести текст.

11.1.4. Что не было учтено?

При построении такого спам-фильтра не были учтены следующие моменты:

- Никак не использована информация, содержащаяся в заголовке письма и адресе отправителя.
- Если слово w не встречается ни в одном из обучающих текстов для какого-то класса, его вероятности $P(w|y)$ сразу оценивается нулем. А значит вероятность того, что текст принадлежит классу y сразу оценивается нулем, что может быть весьма поспешным решением.
- Допустим есть слово w_1, которое не входило в обучающие тексты первого класса (со спамом), и слово w_2, которое не входило в обучающие тексты второго класса (без спама). Тогда, если оба этих слова содержатся в некотором тексте, обе вероятности $P(y_1|spam)$ и $P(y_2|ham)$ будут равны нулю, а значит нельзя будет отнести текст к какому-либо из классов.

11.2. Наивный байесовский классификатор

11.2.1. Байесовский классификатор

Пусть некоторый объект имеет вектор признаков x. Необходимо определить, к какому классу y относится этот объект. Байесовский классификатор $a(x)$ относит объект к такому классу, вероятность которого при условии, что реализовался данный объект, максимальна:

$$a(x) = \text{argmax}_y\, P(y|x) = \text{argmax}_y\, P(x|y)P(y).$$

Здесь была использована теорема Байеса: $P(y|x) = P(x|y)P(y)/P(x)$.

11.2.2. Необходимость использования теоремы Байеса

Непосредственное вычисление $P(y|x)$ заключается в том, что необходимо рассмотреть множество объектов, которые имеют признаковое описание x, и найти долю класса y среди этого множества. Но возможных признаковых описаний огромное количество, а значит вряд ли в обучающей выборке будет достаточное количество объектов для всякого возможного x. Таким образом, не получится вычислять $P(y|x)$ непосредственно и приходится применять теорему Байеса.

Теорема Байеса позволяет переходить к $P(x|y)$, то есть фактически к плотности распределения x при условии класса y (в случае вещественных признаков). Последнюю величину уже можно оценивать по обучающей выборке.

Применение классификатора происходит следующим образом:

$$a(x) = \operatorname{argmax}_y P(x|y)P(y).$$

11.2.3. Проблема нехватки данных

Однако все еще стоит проблема нехватки данных. Пусть, например, обучающая выборка состоит из 100 000 объектов, а пространство признаков имеет размерность 10 000. В этом случае восстановить плотность как функцию от всех признаков достаточно затруднительно.

11.3. Восстановление распределений (часть 1)

Непосредственно восстановить распределение $P(x|y)$ не получается из-за рассмотренной выше проблемы нехватки данных.

11.3.1. Наивный байесовский классификатор

Одно из решений проблемы нехватки данных — использование «наивного» байесовского классификатора, то есть байесовского классификатора:

$$a(x) = \operatorname{argmax}_y P(y|x) = \operatorname{argmax}_y P(x|y)P(y).$$

и «наивной» гипотезы, что плотность распределения расписывается в произведение плотностей по каждому признаку:

$$P(x|y) = P(x_{(1)}|y)P(x_{(2)}|y)...P(x_{(N)}|y),$$

где $x_{(k)}$ — k-ый признак объекта x.

Эта гипотеза выполняется только в случае, если признаки независимые. Это далеко не всегда так, но с некоторой степенью точности таким приближением пользоваться можно.

11.3.2. Восстановление распределений $P(x_{(k)}|y)$

Таким образом, при обучении необходимо определить по обучающей выборке распределения $P(x_{(k)}|y)$ и априорные вероятности классов $P(y)$.

Оценить априорные вероятности классов на основе выборки можно следующим образом:

$$P(y) \approx \frac{\ell_y}{\ell},$$

где ℓ_y — количество объектов класса y в обучающей выборке, а ℓ — размер обучающей выборки. Если соотношение долей классов в обучающей выборке не отражает их реальное соотношение, априорные вероятности классов должны быть взяты из внешних данных.

Распределение $P(x_{(k)}|y)$ можно оценить как долю объектов с данным значением признака $x_{(k)}$ среди объектов класса y:

$$P(x_{(k)}|y) = \frac{1}{\ell_y}\#(x_{(k)}, y).$$

Таким образом, для бинарных признаков:

$$P(x_{(k)} = 0|y) = \frac{1}{\ell_y}\#(x_{(k)} = 0, y), \qquad P(x_{(k)} = 1|y) = \frac{1}{\ell_y}\#(x_{(k)} = 1, y).$$

11.3.3. Пример: классификация текстов

Классификатор текстов можно построить следующим образом. По обучающей выборке строится словарь всех входящих в тексты обучающей выборки слов. Каждый текст будет характеризоваться вектором из бинарных признаков: $x_{(k)} = 1$, если слово w_k присутствует в тексте, а если не присутствует — $x_{(k)} = 0$.

После этого можно восстановить распределения как это описано выше:

$$P(x_{(k)} = 0|y) = \frac{1}{\ell_y}\#(x_{(k)} = 0, y), \qquad P(x_{(k)} = 1|y) = \frac{1}{\ell_y}\#(x_{(k)} = 1, y).$$

После этого можно применить наивный байесовский классификатор и таким образом решить задачу классификации.

Следует обратить внимание, что получается не совсем то, что было в примере со спамом. Предлагается самостоятельно подумать, почему это так.

11.3.4. Сглаживание вероятностей

Если в обучающей выборке среди множества объектов определенного класса y никогда не встречалось какое-то значение t некоторого признака $x_{(k)}$, то вероятность $P(x_{(k)} = t|y) = 0$. Поскольку в выражении, которое требуется максимизировать, стоит произведение таких вероятностей, все это выражение будет равно нулю. Таким образом, любой объект, только на основании того, что значение признака $x_{(k)} = t$, не будет отнесен к классу y, что, вообще говоря, неправильно.

Избежать такой ситуации можно с помощью сглаживания вероятности, например следующим образом:

$$P(x_{(k)} = 1|y) = \frac{\#(x_{(k)} = 1, y) + a}{\ell_y + a + b}, \qquad P(x_{(k)} = 0|y) = \frac{\#(x_{(k)} = 0, y) + b}{\ell_y + a + b}.$$

Константы a и b выбираются таким образом, что качество алгоритма получалось наибольшим.

11.4. Восстановление распределений (часть 2)

Рассмотренный ранее способ восстановления распределений для бинарных признаков не годится в случае вещественных признаков.

11.4.1. Параметрическое восстановление распределения

Однако можно предположить, что распределение имеет какой-то определенный вид: пуассоновское, экспоненциальное или нормальное, и попробовать восстановить его. Это метод называется методом параметрического восстановления распределений.

Пусть, например, рассматривается нормальное распределение:

$$X \sim \mathcal{N}(\mu, \sigma^2).$$

Плотность распределения имеет вид

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

полностью определяется значениями двух параметров: математического ожидания μ и дисперсии σ^2. С помощью оценок максимального правдоподобия для этих параметров:

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} x_i, \qquad \hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \hat{\mu})^2$$

можно оценить параметры по обучающей выборке. Несмещенный вариант оценки для дисперсии:

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{i=1}^{N} (x_i - \hat{\mu})^2 .$$

Другой пример — распределение Бернулли. Это распределение характеризуется одним параметром — вероятность того, что случайная величина принимает значение 1. Этот параметр можно оценить как долю случаев, в которых случайная величина равнялась 1:

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N}[x_i = 1].$$

Также следует отметить, что рассмотренные ранее оценки распределения бинарных признаков — частный случай параметрического восстановления плотности.

11.4.2. Рекомендации по выбору распределений

Верны следующие общие рекомендации по выбору распределении при использовании метода параметрического восстановления:

- Если решается задача, связанная с текстами или какими-то другими разряженными дискретными признаками, то хорошо подходит мультиномиальное распределение.
- Если в задаче есть непрерывные признаки с небольшим разбросом, то можно попробовать использовать нормальное распределение.
- Для непрерывных признаков с большим разбросом нужны более «размазанные», нежели нормальное, распределения.

При этом не обязательно ограничиваться наивным байесовским классификатором. Проблему нехватки данных можно решать с помощью параметрической оценки многомерных распределений, но решение искать в каком-то узком классе так, чтобы оно определялось небольшим числом параметров.

Например, можно предположить, что распределение является многомерным нормальным распределением

$$p(x) = \frac{1}{(2\pi)^{\frac{n}{2}}\sqrt{\det\Sigma}}e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

и по выборке оценивать его параметры: вектор средних μ и матрицу ковариаций Σ. Причем количество вещественных параметров будет гораздо больше, чем в «наивном подходе». Действительно, в «наивном» подходе параметры — это n средних и n дисперсий, а в случае многомерного нормального распределения — вектор размера n и матрица размера $n \times n$. Поэтому из-за нехватки данных оценка каких-то параметров может получиться неверной, а также часто возникают неустойчивые операции, например обращение почти вырожденных матриц и так далее.

Существует также непараметрическое восстановление плотности, о котором будет подробнее рассказано в следующих курсах.

11.5. Минимизация риска

В этом разделе предложен другой взгляд на байесовскую классификацию, а также будет произведено обобщение на случай задачи регрессии.

11.5.1. Байесовская регрессия

Байесовский классификатор определяется выражением:

$$a(x) = \operatorname{argmax}_y P(x|y)P(y),$$

где x — признаковое описание, y — класс.

Применить такую же формулу в случае регрессии (в этом случае y — прогнозируемая величина) не получится, так как вряд ли получится восстановить распределение $P(x|y)$, поскольку y — вещественное число.

Если воспользоваться при решении задачи регрессии выражением:

$$a(x) = \operatorname{argmax}_y P(y|x),$$

то это будет соответствовать поиску максимума функции плотности по y при выбранном x. Не очевидно, что это будет хорошим решением задачи регрессии.

11.5.2. Штрафы за ошибки

Часто бывает необходимо по-разному штрафовать алгоритм за разные типы ошибок. Например, в задаче классификации нефтяных месторождений с двумя классами «есть нефть» и «нет нефти» ошибочный положительный результат — более критичная ошибка, так как бурение скважины требует огромных денежных и временных затрат.

В задачах регрессии штрафы за ошибки еще более естественны: так как искомую зависимость идеально восстановить невозможно, требуется именно минимально отклониться от нее. В задачах регрессии в качестве меры отклонения часто используются квадратичные потери (MSE) и сумма модулей отклонения (MAE):

$$MSE = \frac{1}{\ell}\sum_{i=1}^{\ell}(y_i - a(x_i))^2, \qquad MAE = \frac{1}{\ell}\sum_{i=1}^{\ell}|y_i - a(x_i)|.$$

11.5.3. Более общий подход

Пусть для некоторого объекта x необходимо сделать прогноз $a(x)$. Какая именно задача, задача регрессии или классификации, рассматривается, не имеет значения. Пусть также y — правильный ответ, а функция $L(y, a(x))$ определяет величину ошибки алгоритма и задается в зависимости от рассматриваемой задачи и желаемых свойств алгоритма.

В задаче классификации можно использовать в качестве функции $L(y, a(x))$ индикатор того, что точный ответ y не совпадает с прогнозом $a(x)$:

$$L(y, a(x)) = [y \neq a(x)].$$

Такой выбор функции приведет к тому, что полученный классификатор будет уже рассмотренным ранее байесовским классификатором, но об этом будет рассказано позднее.

В задаче регрессии используется квадратичная функция:

$$L(y, a(x)) = (y - a(x))^2.$$

11.5.4. Оптимальный байесовский классификатор

Так называемый функционал риска $R(a, x)$ определяется как условное математическое ожидание потерь при известном x и ответе алгоритма a:

$$R(a, x) = \mathbb{E}\left(L(y, a(x))\middle|x\right).$$

Можно строить ответы алгоритма таким образом, чтобы минимизировать ожидаемые потери:

$$a(x) = \operatorname{argmin}_s R(s, x).$$

В случае задачи классификации можно записать следующее:

$$R(s, x) = \mathbb{E}\left(L(y, s)\middle|x\right) = \sum_{y \in Y} L(y, s) P(y|x)$$

$$a(x) = \operatorname{argmin}_s R(s, x) = \operatorname{argmin}_s \sum_{y \in Y} L(y, s) P(y|x) = \operatorname{argmin}_s \sum_{y \in Y} L(y, s) P(y) P(x|y).$$

Такой классификатор называется оптимальным байесовским классификатором, потому что он минимизирует ожидаемые потери. Реальный классификатор, конечно, не будет оптимальным из-за использования вероятностных оценок, а не истинных вероятностей.

11.5.5. Оптимальный байесовский регрессор

Для задачи регрессии выражения выглядят аналогичным образом:

$$R(s, x) = \mathbb{E}\left(L(y, s)\middle|x\right) = \int_{y \in Y} L(y, s) P(y|x) dy$$

$$a(x) = \operatorname{argmin}_s R(s, x) = \operatorname{argmin}_s \int_{y \in Y} L(y, s) P(y|x) dy.$$

На практике данный результат используется не для решения задачи регрессии, а чтобы проанализировать разные функции потерь, о чем будет рассказано позднее.

11.5.6. Функционал среднего риска

Функционал среднего риска
$$R(a) = \mathbb{E}_x R(a(x), x)$$
позволяет оценить, насколько хорошо работает алгоритм в среднем, а не для конкретного x.

Для определенности дальше рассматривается случай задачи классификации объектов с дискретными признаками. Остальные случаи, например случаи задачи регрессии или задачи классификации объектов с непрерывными признаками, полностью аналогичны с точностью до замены суммы на интеграл.

В данном ситуации функционал среднего риска просто представляется взвешенной суммой возможных значений функционала риска, где в качестве весов выступают вероятности $P(x)$:
$$R(a) = \sum_{x \in X} R(a(x), x) P(x).$$
Поскольку $R(s, x) \geq \min_s R(s, x)$, верна следующая оценка снизу для $R(a)$:
$$R(a) = \sum_{x \in X} R(a(x), x) P(x) \geq \sum_{x \in X} \min_s R(s, x) P(x).$$
Эта нижняя оценка достигается, если $R(s, x) = \min_s R(s, x)$, то есть в уже знакомом случае оптимального байесовского классификатора. Таким образом, оптимальный байесовский классификатор минимизирует не только функционал риска, но и функционал среднего риска.

11.6. Минимизация риска и анализ функции потерь

Итак, уже в прошлом видео было анонсировано, что с помощью нового, более общего взгляда на байесовскую классификацию можно получить несколько интересных результатов.

11.6.1. Оптимальный байесовский классификатор

Можно показать, что оптимальный байесовский классификатор:
$$a(x) = \operatorname{argmin}_s R(s, x) = \operatorname{argmin}_s \sum_{y \in Y} L(y, s) P(y|x),$$
в случае, если функция потерь равна индикатору того, что прогноз алгоритма $a(x)$ не совпал с правильным ответом y:
$$L(y, a(x)) = [y \neq a(x)],$$
переходит в знакомый с начала урока байесовский классификатор.

Действительно, если подставить функцию потерь в минимизируемое выражение:
$$\sum_{y \in Y} L(s, y) P(y|x) = \sum_{y \in Y \setminus \{s\}} P(y|x) = \sum_{y \in Y} P(y|x) - P(s|x) \to \min_s \quad \Longrightarrow \quad P(s|x) \to \max_s,$$
Таким образом, классификатор имеет вид:
$$a(x) = \operatorname{argmax}_y P(y|x) = \operatorname{argmax}_y P(y) P(x|y),$$
то есть является байесовским классификатором.

11.6.2. Квадратичная функция потерь в регрессии

Оказывается, такой подход годится для анализа различных функций потерь в задаче регрессии. Например, в случае квадратичной функции потерь:
$$\int_Y (t - y)^2 p(y|x) dy \to \min_t.$$

Минимум можно найти, если приравнять производную по t к нулю:

$$\frac{\partial}{\partial t}\int_Y (t-y)^2 p(y|x)dy = 2\int_Y (t-y)p(y|x)dy = 2\left(t\int_Y p(y|x)dy - \int_Y yp(y|x)dy\right) = 0.$$

Так как $p(y|x)$ — плотность вероятности, $\int_Y p(y|x)dy = 1$:

$$a(x) = t = \int_Y yp(y|x)dy = \mathbb{E}(y|x).$$

Таким образом, прогноз алгоритма должен равняться условному математическому ожиданию $\mathbb{E}(y|x)$.

11.6.3. Абсолютное отклонение

В случае, когда функция потерь — абсолютное отклонение:

$$\int_Y |t-y|p(y|x)dy \to \min_t,$$

выкладки производятся точно также. Единственный нюанс заключается в том, что модуль не имеет производной в нуле, поэтому точку $y = t$ следует заблаговременно исключить из области интегрирования (важно, что это не изменит значение интеграла):

$$\frac{\partial}{\partial t}\int_Y |t-y|p(y|x)dy = \frac{\partial}{\partial t}\int_{Y\setminus\{t\}} |t-y|p(y|x)dy = \int_{Y\setminus\{t\}} \operatorname{sign}(t-y)p(y|x)dy =$$
$$= \int_{\{t>y\}} p(y|x)dy - \int_{\{t<y\}} p(y|x)dy = P(\{t>y\}|x) - P(\{t<y\}|x) = 0.$$

Таким образом, учитывая, что $P(\{t=y\}|x) = 0$, можно получить:

$$P(\{t>y\}|x) = P(\{t<y\}|x) = \frac{1}{2}.$$

Другими словами, ответ алгоритма оценивает $1/2$ квантиль (медиану).

11.6.4. Оценка вероятности

Рассматривается задача бинарной классификации $Y = \{0, 1\}$. Необходимо, чтобы алгоритм классификации оценивал вероятность того, что объект принадлежит к первому классу $p = P(1|x)$. Оказывается, что получить требуемый результат можно, выбрав в качестве функции потерь так называемую функцию Log loss:

$$L(y, a(x)) = -y\ln a(x) - (1-y)\ln(1-a(x))$$

Условие минимальности потерь тогда принимает вид:

$$\sum_{y\in Y}\Big(-y\ln t - (1-y)\ln(1-t)\Big)P(y|x) = -(1-p)\ln(1-t) - p\ln t \to \min_t,$$

где использовано обозначение $p = P(1|x)$. Минимум можно найти вычислением производной по t:

$$\frac{\partial}{\partial t}\big(-(1-p)\ln(1-t) - p\ln t\big) = \frac{1-p}{1-t} - \frac{p}{t} = \frac{t-p}{(1-t)t} = 0.$$

Получается требуемый результат:

$$a(x) = t = p,$$

то есть ответ алгоритма t должен равняться вероятности p того, что объект принадлежит к первому классу.

11.6.5. Обоснование метода анализа функции потерь

Следует напомнить, что в байесовской классификации минимизируется именно функционал среднего риска:

$$R(a) = \mathbb{E}_{x,y} L(y, a(x)).$$

Поскольку ошибка Q на обучающей выборке является эмпирической оценкой функционала среднего риска:

$$Q = \frac{1}{\ell} \sum_{i=1}^{\ell} L(y_i, a(x_i)) \sim \mathbb{E}_{x,y} L(y, a(x)),$$

результаты приведенного выше метода анализа функции потерь остаются верны не только в случае использования байесовского классификатора или байесовской регрессии, но и для произвольного метода решения, в ходе которого минимизируется ошибка на обучающей выборке.