Скользящий контроль

Материал из MachineLearning.

(Перенаправлено с Кросс-валидация)

Скользящий контроль или **кросс-проверка** или **кросс-валидация** (cross-validation, CV) — процедура эмпирического оценивания обобщающей способности алгоритмов, обучаемых по прецедентам.

Фиксируется некоторое множество разбиений исходной выборки на две подвыборки: *обучающую* и *контрольную*. Для каждого разбиения выполняется настройка алгоритма по обучающей подвыборке, затем оценивается его средняя ошибка на объектах контрольной подвыборки. *Оценкой скользящего контроля* называется средняя по всем разбиениям величина ошибки на контрольных подвыборках.

Если выборка независима, то средняя ошибка *скользящего контроля* даёт несмещённую оценку вероятности ошибки. Это выгодно отличает её от средней ошибки на обучающей выборке, которая может оказаться смещённой (оптимистически заниженной) оценкой вероятности ошибки, что связано с явлением переобучения.

Скользящий контроль является стандартной методикой тестирования и сравнения алгоритмов классификации, регрессии и прогнозирования.

Содержание

Определения и обозначения

Рассматривается задача обучения с учителем.

Пусть X — множество описаний объектов, Y — множество допустимых ответов.

Задана конечная выборка прецедентов $X^L = (x_i, y_i)_{i=1}^L \subset X \times Y$.

Задан алгоритм обучения — отображение μ, которое произвольной конечной выборке прецедентов X^m ставит в соответствие функцию (алгоритм) $a\colon X \to Y$.

Качество алгоритма a оценивается по произвольной выборке прецедентов X^m с помощью *функционала качества* $Q(a, X^m)$. Для процедуры скользящего контроля не важно, как именно вычисляется этот функционал. Как правило, он аддитивен по объектам выборки:

$$Q(a, X^m) = \frac{1}{m} \sum_{x_i \in X^m} \mathcal{L}(a(x_i), y_i),$$

где $\mathcal{L}(a(x_i), y_i)$ — неотрицательная функция потерь, возвращающая величину ошибки ответа алгоритма $a(x_i)$ при правильном ответе y_i.

Процедура скользящего контроля

Выборка X^L разбивается N различными способами на две непересекающиеся подвыборки: $X^L = X_n^m \cup X_n^k$, где X_n^m — обучающая подвыборка длины m, X_n^k — контрольная подвыборка длины $k = L - m$, $n = 1, \ldots, N$ — номер разбиения.

Для каждого разбиения n строится алгоритм $a_n = \mu(X_n^m)$ и вычисляется значение функционала качества $Q_n = Q(a_n, X_n^k)$. Среднее арифметическое значений Q_n по всем разбиениям называется *оценкой скользящего контроля*:

$$CV(\mu, X^L) = \frac{1}{N} \sum_{n=1}^{N} Q(\mu(X_n^m), X_n^k).$$

Различные варианты скользящего контроля отличаются видами функционала качества и способами разбиения выборки.

Доверительное оценивание

Кроме среднего значения качества на контроле строят также доверительные интервалы.

Непараметрическая оценка доверительного интервала. Строится вариационный ряд значений $Q_n = Q(a_n, X_n^k)$, $n = 1,\ldots,N$:

$$Q^{(1)} \le Q^{(2)} \le \ldots \le Q^{(N)}.$$

Утверждение 1. Если разбиения осуществлялись случайно, независимо и равновероятно, то с вероятностью $\eta = \frac{t}{N+1}$ значение случайной величины $Q(a(X^m), X^k)$ не превосходит $Q^{(N-t+1)}$.

Следствие 1. Значение случайной величины $Q(a(X^m), X^k)$ не превосходит $Q^{(N)}$ с вероятностью $\eta = \frac{1}{N+1}$.

В частности, для получения верхней оценки с надёжностью 95% достаточно взять $N = 20$ разбиений.

Утверждение 2. Если разбиения осуществлялись случайно, независимо и равновероятно, то с вероятностью $\eta = \frac{2t}{N+1}$ значение случайной величины $Q(a(X^m), X^k)$ не выходит за границы доверительного интервала $[Q^{(t)}, Q^{(N-t+1)}]$.

Следствие 2. Значение случайной величины $Q(a(X^m), X^k)$ не выходит за границы вариационного ряда $[Q^{(1)}, Q^{(N)}]$ с вероятностью $\eta = \frac{2}{N+1}$.

В частности, для получения двусторонней оценки с надёжностью 95% достаточно взять $N = 40$ разбиений.

Параметрические оценки доверительного интервала основаны на априорном предположении о виде распределения случайной величины $Q(a(X^m), X^k)$. Если априорные предположения не выполняются, доверительный интервал может оказаться сильно смещённым. В частности, если предположения о нормальности распределения не выполнены, то нельзя пользоваться стандартным «правилом двух сигм» или «трёх сигм». Джон Лангфорд в своей диссертации (2002) указывает на распространённую ошибку, когда правило двух сигм применяется к функционалу частоты ошибок, имеющему на самом деле биномиальное распределение. Однако биномиальным распределением в общем случае тоже пользоваться нельзя, поскольку в результате обучения по случайным подвыборкам X^m вероятность ошибки алгоритма $a(X^m)$ оказывается случайной величиной. Следовательно, случайная величина $Q(a(X^m), X^k)$ описывается не биномиальным распределением, а (неизвестной) смесью биномиальных распределений. Аппроксимация смеси биномиальным распределением может приводить к ошибочному сужению доверительного интервала. Приведённые выше непараметрические оценки лишены этого недостатка.

Стратификация

Стратификация выборки — это способ уменьшить разброс (дисперсию) оценок скользящего контроля, в результате чего получаются более узкие доверительные интервалы и более точные (tight) верхние оценки.

Стратификация заключается в том, чтобы заранее поделить выборку на части (страты), и при разбиении на обучение длины *m* и контроль длины *k* гарантировать, что каждая страта будет поделена между обучением и контролем в той же пропорции $m{:}k$.

Стратификация классов в задачах классификации означает, что каждый класс делится между обучением и контролем в пропорции $m{:}k$.

Стратификация по вещественному признаку. Объекты выборки сортируются согласно некоторому критерию, например, по возрастанию одного из признаков. Затем выборка разбивается на *k* последовательных страт одинаковой (с точностью до 1) длины. При формировании контрольных выборок из каждой страты выбирается по одному объекту, либо с заданным порядковым номером внутри страты, либо случайным образом.

Разновидности скользящего контроля

Возможны различные варианты скользящего контроля, отличающиеся способами разбиения выборки.

Полный скользящий контроль (complete CV)

Оценка скользящего контроля строится по всем $N = C_L^k$ разбиениям. В зависимости от k (длины обучающей выборки) различают:

- Частный случай при $k = 1$ — контроль по отдельным объектам (leave-one-out CV);

Было показано (Li,1987) , что контроль по отдельным объектом является асимптотически оптимальным при некоторых условиях[1], то есть:

$\frac{L_n(\widehat{m})}{\inf_{m \in M_n} L_n(m)} \to 1$ по вероятности,

где:

1. M_n - класс сравниваемых моделей;
2. $L_n(m)$ - среднеквадратичная ошибка при выборе m-ой модели;
3. $\widehat{m} = \arg\min_{m \in M_n} CV(m)$.

- Общий случай при $k>2$. Здесь число разбиений $N = C_L^k$ становится слишком большим даже при сравнительно малых значениях k, что затрудняет практическое применение данного метода. Для этого случая *полный скользящий контроль* используется либо в теоретических исследованиях (Воронцов,2004), либо в тех редких ситуациях, когда для него удаётся вывести эффективную вычислительную формулу. Например, такая формула известна для метода *k* ближайших соседей [1], что позволяет эффективно выбирать параметр *k*. На практике чаще применяются другие разновидности *скользящего контроля*.

Случайные разбиения

Разбиения $n = 1,\ldots,N$ выбираются случайно, независимо и равновероятно из множества всех C_L^k разбиений. Именно для этого случая справедливы приведённые выше оценки доверительных интервалов. На практике эти оценки, как правило, без изменений переносится и на другие способы разбиения выборки.

Контроль на отложенных данных (hold-out CV)

Оценка скользящего контроля строится по одному случайному разбиению, $N = 1$.

Этот способ имеет существенные недостатки:

1. Приходится слишком много объектов оставлять в контрольной подвыборке. Уменьшение длины обучающей подвыборки приводит к смещённой (пессимистически завышенной) оценке вероятности ошибки.
2. Оценка существенно зависит от разбиения, тогда как желательно, чтобы она характеризовала только алгоритм обучения.
3. Оценка имеет высокую дисперсию, которая может быть уменьшена путём усреднения по разбиениям.

Следует различать скользящий контроль по отложенным данным и контроль по тестовой выборке. Если во втором случае оценивается вероятность ошибки для классификатора, построенного по обучающей подвыборке, то в первом случае - для классификатора, построенного по полной выборке (то есть доля ошибок вычисляется не для того классификатора, который выдается в качестве результата решения задачи).

Контроль по отдельным объектам (leave-one-out CV)

Является частным случаем полного скользящего контроля при $k = 1$, соотвественно, $N = L$. Это, пожалуй, самый распространённый вариант скользящего контроля.

Преимущества LOO в том, что каждый объект ровно один раз участвует в контроле, а длина обучающих подвыборок лишь на единицу меньше длины полной выборки.

Недостатком LOO является большая ресурсоёмкость, так как обучаться приходится L раз. Некоторые методы обучения позволяют достаточно быстро перенастраивать внутренние параметры алгоритма при замене одного обучающего объекта другим. В этих случаях вычисление LOO удаётся заметно ускорить.

Контроль по *q* блокам (*q*-fold CV)

Выборка случайным образом разбивается на *q* непересекающихся блоков одинаковой (или почти одинаковой) длины $k_1,\ldots,k_q$:

$$X^L = X_1^{k_1} \cup \ldots \cup X_q^{k_q},$$

$k_1 + \ldots + k_q = L$. Каждый блок по очереди становится контрольной подвыборкой, при этом обучение производится по остальным $q-1$ блокам. Критерий определяется как средняя ошибка на контрольной подвыборке:

$$CV(\mu, X^L) = \frac{1}{q} \sum_{n=1}^{q} Q\big(\mu(X^L \setminus X_n^{k_n}), X_n^{k_n}\big).$$

Это компромисс между LOO, hold-out и случайными разбиениями. С одной стороны, обучение производится только *q* раз вместо *L*. С другой стороны, длина обучающих подвыборок, равная $L\frac{q-1}{q}$ с точностью до округления, не сильно отличается от длины полной выборки *L*. Обычно выборку разбивают случайным образом на 10 или 20 блоков.

Контроль по *r*×*q* блокам (*r*×*q*-fold CV)

Контроль по *q* блокам (*q*-fold CV) повторяется *r* раз. Каждый раз выборка случайным образом разбивается на *q* непересекающихся блоков. Этот способ наследует все преимущества *q*-fold CV, при этом появляется дополнительная возможность увеличивать число разбиений.

Данный вариант скользящего контроля, со стратификацией классов, является стандартной методикой тестирования и сравнения алгоритмов классификации. В частности, он применяется в системах WEKA и «Полигон алгоритмов».

Скользящий контроль в задачах прогнозирования

В задачах прогнозирования, динамического обучения, обучения с подкреплением и активного обучения прецеденты изначально линейно упорядочены по времени их появления. В этом случае варианты скользящего контроля не так разнообразны.

Контроль при нарастающей длине обучения

Обучающая подвыборка образуется всеми *прошлыми* объектами $X_n^m = \{x_1,\ldots,x_n\}$. Контрольная подвыборка образуется всеми *будущими* объектами $X_n^k = \{x_{n+\delta+1},\ldots,x_{n+\delta+k}\}$, где $\delta \geq 0$ — величина задержки прогнозирования (обычно $\delta = 0$). Момент «настоящего времени» *n* «скользит» по выборке данных:

$$CV(\mu, X^L) = \frac{1}{T_2 - T_1 + 1} \sum_{n=T_1}^{T_2} Q(\mu(X_n^m), X_n^k),$$

где T_1 — минимальная длина обучающей выборки, требуемая для нормальной работы алгоритма обучения μ, $T_2 = L - \delta - k$.

Поскольку длина обучения $m = n$ увеличивается со временем, точность прогнозов может постепенно улучшаться. Этот побочный эффект является нежелательным, если скользящий контроль применяется для оценивания качества алгоритма обучения.

Контроль при фиксированной длине обучения

Отличается от предыдущего варианта только тем, что длина обучения *m* фиксируется, ограничивась последними *m* прецедентами ряда: $X_n^m = \{x_{n-m+1},\ldots,x_n\}$. При этом полагают $T_1 = m$.

Недостатки скользящего контроля

1. Задачу обучения приходится решать *N* раз, что сопряжено со значительными вычислительными затратами.

2. Оценка скользящего контроля предполагает, что алгоритм обучения μ уже задан. Она ничего не говорит о том, какими свойствами должны обладать «хорошие» алгоритмы обучения, и как их строить. Такого рода подсказки дают, например, теоретические оценки обобщающей способности.
3. Попытка использовать скользящий контроль для обучения, в роли оптимизируемого критерия, приводит к тому, что он утрачивает свойство несмещённости, и снова возникает риск переобучения.
4. Скользящий контроль дает несмещенную точечную, но не интервальную оценку риска. В настоящее время не существует методов построения на основе скользящего контроля точных доверительных интервалов для риска, то есть математического ожидания потерь (в частности, вероятности ошибочной классификации).

Применения скользящего контроля

На практике скользящий контроль применяется для оптимизации некоторых критически важных параметров, как правило, определяющих структуру или сложность используемой модели алгоритма, и имеющих относительно небольшое число возможных значений.

Примеры:

- Выбор модели алгоритмов из небольшого множества альтернативных вариантов.
- Оптимизация параметра регуляризации, в частности:
  - параметра регуляризации в гребневой регрессии;
  - параметра сокращения весов (weight decay) в нейронных сетях;
  - параметра C в методе опорных векторов.
- Оптимизация ширины окна в методах
  - парзеновского окна;
  - ближайшего соседа;
  - ядерного сглаживания.
- Оптимизация числа нейронов в скрытом слое многослойной нейронной сети.
- Выбор информативного набора признаков.
- Редукция решающего дерева.
- Структурная минимизация риска.

Примечания

Литература

1. *Воронцов К. В.* Комбинаторный подход к оценке качества обучаемых алгоритмов. — Математические вопросы кибернетики / Под ред. О. Б. Лупанов. — М.: Физматлит, 2004. — Т. 13. — С. 5–36.
2. *Эфрон Б.* Нетрадиционные методы многомерного статистического анализа. — М: Финансы и статистика. — 1988.
3. *Langford J.* Quantitatively Tight Sample Complexity Bounds (http://citeseer.ist.psu.edu/langford02quantitatively.html) . — Carnegie Mellon Thesis. — 2002. — 124 с.
4. *Kohavi R.* A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection (http://citeseer.ist.psu.edu/kohavi95study.html) . — 14th International Joint Conference on Artificial Intelligence, Palais de Congres Montreal, Quebec, Canada. — 1995. — С. 1137-1145.
5. *Mullin M., Sukthankar R.* Complete Cross-Validation for Nearest Neighbor Classifiers (http://citeseer.ist.psu.edu/309025.html) . — Proceedings of International Conference on Machine Learning. — 2000. — С. 1137-1145.

*Это незавершённая статья. Вы поможете проекту, исправив и дополнив (http://www.machinelearning.ru/wiki/index.php?title=%D0%A1%D0%BA%D0%BE%D0%BB%D1%8C%D0%B7%D1%8F%D1%89%D0%B8%D0%B9_%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%l
её.*

Категории: Незавершённые статьи | Машинное обучение
