Семинар 11.
ММП, весна 2013
30 апреля

Илья Толстихин
iliya.tolstikhin@gmail.com

Темы семинара:

- Деревья для регрессии и классификации;
- Критерии ветвления деревьев;
- Прунинг деревьев;
- Cost-complexity pruning.

Материалы первой части этого семинара вам должны быть частично знакомы по лекциям. Полезным здесь является определение меры качества вершины в дереве классификации — Gini Index.

1 Наращивание деревьев

1.1 Регрессия

Рассмотрим задачу регрессии и будем решать ее с помощью дерева регрессии (про деревья подробно было рассказано на лекциях).

Задача. *Предположим, что зафиксировано разбиение пространства объектов* $\mathbb{X} = \mathbb{R}^d$ *на непересекающиеся области* R_m, $m = 1, \ldots, M$, *где* M *— число листовых вершин некоторого дерева* T. *Как следует расставить ответы* $c_m \in \mathbb{R}$ *для каждой области* R_m, *чтобы полученное дерево минимизировало функционал*

$$R(T) = \sum_{i=1}^{\ell} \big(y_i - T(x_i)\big)^2 ? \tag{RSS}$$

Несложно доказать, что для этого в качестве ответа в вершине нужно использовать среднее значение ответов попавших в эту вершину объектов обучающей выборки:

$$c_m = \frac{1}{N_m} \sum_{x_i} [x_i \in R_m] y_i.$$

Весь вопрос остается в том, как выбрать все прочие параметры дерева — то есть как его нарастить. Поскольку задача поиска оптимального дерева является в общем

случае задачей NP-полной, мы будем пользоваться жадной стратегией. А именно: начиная от всей обучающей выборки, мы будем последовательно дробить каждую из текущих областей на две части с помощью нехитрых бинарных предикатов вида $\varphi(X) = [X_j \leqslant s]$, где номер признака $j \in \{1, \ldots, d\}$ и порог s являются параметрами закономерности. Дерево в узлах которого находятся такого вида бинарные предикаты задает разбиение пространства объектов на области со сторонами параллельными осям координат. Теперь весь вопрос в том, как мы будем на каждом шаге для фиксированного подмножества обучающей выборки (попавшего в рассматриваемую область, задаваемую некоторой листовой вершиной текущего дерева) выбирать номер признака j и порог s. Мы, конечно, хотим минимизировать функционал (RSS). Поэтому нам остается решить задачу

$$(j, s) = \arg\min_{j,s} \left[\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right],$$

где

$$R_1(s, j) = \{X : X_j \leqslant s\}, \quad R_2(s, j) = \{X : X_j \leqslant s\}.$$

Оптимизацию внутри по c_1 и c_2 мы уже умеем решать. Все остальное можно решить перебором признаков и конечного числа значений порогов для каждого признака.

1.2 Классификация

Рассмотрим теперь задачу классификации с $\mathbb{Y} = \{1, \ldots, K\}$. В случае наращивания деревьев для классификации квадратичные потери нам уже не подходят. Пусть у нас есть готовое дерево T, и область, соответствующая его m-й листовой вершине, снова обозначим R_m. Пусть в m-ю листовую вершину попало ровно N_m объектов обучающей выборке. Введем обозначение:

$$\hat{p}_{m,k} = \frac{1}{N_m} \sum_{i=1}^{\ell} [x_i \in R_m][y_i = k],$$

то есть это доля объектов класса k среди объектов, попавших в листовую вершину m. Пользуясь этим обозначением, определим разметку листовых вершин:

$$k(m) = \arg\max_{k \in \mathbb{Y}} \hat{p}_{m,k}.$$

Перечислим функционалы качества листовых вершин, часто используемые при наращивании деревьев классификации (чем меньше значение, тем лучше):

- Ошибка классификации:
$$1 - \hat{p}_{m,k(m)},$$
- Gini index:
$$\sum_{k=1}^{K} \hat{p}_{m,k}(1 - \hat{p}_{m,k}),$$

- Кросс-энтропия:

$$-\sum_{k=1}^{K} \hat{p}_{m,k} \log \hat{p}_{m,k}.$$

Задача. *Предположим, что по обучающей выборке мы нарастили дерево T и посчитали все значения $\hat{p}_{m,k}$, $k = 1, \ldots, K$, $m = 1, \ldots, M$. Предположим, что построенное дерево будет работать с рандомизацией: на объекте x, попавшем в m-ю вершину, дерево будет отвечать случайной меткой класса k с вероятностью $\hat{p}_{m,k}$:*

$$T(x) = \sum_{m=1}^{M} [x \in R_m] \xi_m, \quad \mathrm{P}(\xi_m = k) = \hat{p}_{m,k}.$$

Докажите, что математическое ожидание частоты ошибки вершины m на объектах обучающей выборки в точности равно значению Gini index m-й вершины.

Если мы снова прибегнем к жадной стратегии наращивания дерева, одним из критериев выбора номера признака j и порога s может служить следующий:

$$(j, s) = \arg\min_{j,s} \left[N_\ell \sum_{k=1}^{K} \hat{p}_{m_\ell,k}(1 - \hat{p}_{m_\ell,k}) + N_r \sum_{k=1}^{K} \hat{p}_{m_r,k}(1 - \hat{p}_{m_r,k}) \right],$$

где m_r и m_ℓ — соответственно правая и левая дочерние вершины текущей вершины, которую мы хотим ветвить, а N_r и N_ℓ — число объектов обучающей выборки, попавших в эти дочерние вершины.

2 Прунинг деревьев

В прошлом разделе мы вкратце ознакомились с методами наращивания деревьев для задач регрессии или классификации. Несложно проверить, что большое дерево T, построенные описанным образом, склонно к переобучению. Самое время познакомиться с одним из методов *пост-прунинга* (post-pruning), задача которых заключается в выборе поддерева T' дерева T с тем же корнем, имеющего лучшее качество работы. Отметим, что пост-прунинг часто предпочтительнее альтернативной стратегии *раннего останова* (early stop), когда мы заканчиваем наращивание дерева при достижении определенного критерия останова. Это связано с жадностью использованной стратегии наращивания дерева: может случиться, что дерево, которое мы получим через пару шагов наращивания, имеет резко выросшее значение функционала качества по сравнению с текущим.

Мы рассмотрим один из классических методов прунинга, предложенный в [1].

2.1 Cost-complexity pruning

Обозначим большое дерево, полученное в результате жадного наращивания, с помощью T_0. Пусть $R(T)$ — функционал качества дерева T, использовавшийся при построении T_0. Отметим, что $R(T)$ вычисляется как сумма вкладов каждой листовой вершины T. Назовем *размером* дерева T число его листовых вершин: $|T|$. Идея метода заключается в выборе поддерева $T \subseteq T_0$ с тем же корнем, минимизирующего следующее выражение:

$$R_\alpha(T) = R(T) + \alpha |T|, \qquad (*)$$

где $\alpha \geqslant 0$ — заранее заданный параметр.

Задача 1. *Докажите, что* $\arg\min\limits_{T \subseteq T_0} R_0(T) = T_0$.

Мы сейчас покажем, что существует последовательность вложенных деревьев с одинаковыми корнями:

$$T_K \subset T_{K-1} \subset \cdots \subset T_0,$$

(где T_K — тривиальное дерево, состоящее из корня дерева T_0), такая что каждое дерево T_i минимизирует критерий (*) для α из интервала $\alpha \in [\alpha_i, \alpha_{i+1})$, причем

$$0 = \alpha_0 < \alpha_1 < \cdots < \alpha_K < \infty.$$

Возможно, существует несколько поддеревьев дерева T_0 с одинаковым значением критерия $R_\alpha(T)$. В этом случае *α-оптимальным* мы будем называть то из них, которое является поддеревом всех остальных (если такое имеется), и будем обозначать его $T_0(\alpha)$.

Рассмотрим любое бинарное дерево T, состоящее из более чем одной вершины, и для любой его внутренней вершины t обозначим с помощью T_t поддерево дерева T с корнем в t. *Левым поддеревом* вершины t будем называть поддерево дерева T с корнем в левой дочерней вершине вершины t, а *правым поддеревом* — с корнем в правой. Будем обозначать их $T_{t,\ell}$ и $T_{t,r}$ соответственно.

Введем следующую величину:

$$g(t, T) = \frac{R(t) - R(T_t)}{|T_t| - 1}.$$

Сразу же отметим, что $g(t, T) > \alpha$ тогда и только тогда, когда $R_\alpha(t) > R_\alpha(T_t)$. Отметим также (это нам пригодится позже), что критерий $R_\alpha(T)$ можно представить в виде суммы $R(t) + \alpha$ всех листовых вершин t дерева T.

Прунинг (сокращение) вершины t заключается в замене поддерева T_t на листовую вершину t.

Докажем ряд утверждений. (Доказательства взяты из [2]).

Теорема 2.1 *Пронумеруем вершины дерева T так, что любая вершина имеет номер меньший, чем ее родительская вершина. Если мы будем посещать вершины в порядке этой нумерации (то есть «снизу вверх») и сокращать текущую вершину t, если $R_\alpha(t) \leqslant R_\alpha(T_t)$, то в результате мы получим дерево $T(\alpha)$ — α-оптимальное поддерево дерева T.*

Доказательство:
Предположим, что мы сейчас находимся в вершине t дерева T, совершив перед этим определенное число шагов (то есть, возможно, сократив некоторое количество вершин). Текущее дерево обозначим с помощью T'. Докажем по индукции, что после завершения текущей итерации — то есть после сокращение вершины t, если $R_\alpha(t) \leqslant R_\alpha\big((T')_t\big)$, — получившееся поддерево с корнем в t является α-оптимальными. В качестве базы индукции рассмотрим внутреннюю вершину, обе дочерних вершины которой являются листовыми. В этом случае утверждение очевидно.

Пусть оба поддерева $(T')_{t,\ell}$ и $(T')_{t,r}$ вершины t являются α-оптимальными. Предположим, что существует некоторое нетривиальное поддерево $(T'')_t$ с корнем в t,

отличное от $(T')_t$, такое что $R_\alpha\big((T'')_t\big) < R_\alpha\big((T')_t\big)$. Поскольку функционал R_α считается как сумма вкладов листовых вершин, мы приходим к выводу, что либо

$$R_\alpha\big((T'')_{t,\ell}\big) < R_\alpha\big((T')_{t,\ell}\big)$$

либо

$$R_\alpha\big((T'')_{t,r}\big) < R_\alpha\big((T')_{t,r}\big),$$

а значит одно из текущих поддеревьев не является α-оптимальным, то есть получаем противоречие. Значит, из всех нетривиальных (содержащих по крайней мере 3 вершины) поддеревьев с корнем в вершине t текущее дерево $(T')_t$ имеет наименьшее значение функционала R_α.

Предположим теперь, что существует некоторое нетривиальное поддерево $(T'')_t$, такое что $R_\alpha\big((T'')_t\big) = R_\alpha\big((T')_t\big)$. В этом случае из аналогичных соображений можно сделать вывод, что непременно

$$R_\alpha\big((T'')_{t,\ell}\big) = R_\alpha\big((T')_{t,\ell}\big)$$

и

$$R_\alpha\big((T'')_{t,r}\big) = R_\alpha\big((T')_{t,r}\big).$$

А поскольку поддеревья $(T')_{t,\ell}$ и $(T')_{t,r}$ оптимальны, то по определению $(T')_{t,\ell}$ является поддеревом дерева $(T'')_{t,\ell}$ и $(T')_{t,r}$ является поддеревом дерева $(T'')_{t,r}$.

Теперь должно быть очевидно, что после принятия решения о сокращении вершины t оставшееся поддерево с корнем в t будет α-оптимальным. То же самое справедливо и для момента, когда мы доберемся до корня всего дерева T, а значит в результате описанной процедуры мы получим $T(\alpha)$. □

Описанный в формулировке теоремы метод назовем *α-отсечением*. Итак, с помощью α-отсечения мы можем получить α-оптимальное поддерево дерева T.

Теорема 2.2 *Обозначим наименьшее значение $g(t,T)$ для всех внутренних вершин дерева T с помощью α_1. Тогда для всех $\alpha < \alpha_1$ α-оптимальным поддеревом дерева T является оно само. Дерево $T_1 = T(\alpha_1)$ получается из дерева T сокращением всех вершин t, для которых $g(t,T) = \alpha_1$. Более того, для всех внутренних вершин t дерева T_1 выполнено $g(t,T_1) > \alpha_1$.*

Доказательство:
Заметим, что если

$$g\big(t,(T)\big) = \frac{R(t) - R(T_t)}{|T_t| - 1} = \alpha_1 > \alpha,$$

то

$$R_\alpha(t) > R_\alpha(T_t),$$

а значит для таких значений α ни одно сокращение внутренней вершины произведено не будет. То есть первых два утверждения теоремы очевидны.

Обратим внимание, что когда мы сокращаем вершины со значением α_1, то значения $R_{\alpha_1}(T_t)$ не меняются для всех вершин, оставшихся в дереве T_1. Мы хотим доказать, что

$$\frac{R(t) - R\big((T_1)_t\big)}{|(T_1)_t| - 1} > \alpha_1$$

для всех внутренних вершин t дерева T_1. Запишем:

$$R(t) - R\big((T_1)_t\big) = R(t) + \alpha_1 - \Big(R\big((T_1)_t\big) + \alpha_1|(T_1)_t|\Big) + \alpha_1\big(|(T_1)_t| - 1\big) =$$
$$= R_{\alpha_1}(t) - R_{\alpha_1}\big((T_1)_t\big) + \alpha_1\big(|(T_1)_t| - 1\big) =$$
$$= R_{\alpha_1}(t) - R_{\alpha_1}(T_t) + \alpha_1\big(|(T_1)_t| - 1\big) > \alpha_1\big(|(T_1)_t| - 1\big),$$

где мы воспользовались тем фактом, что раз вершина t осталась внутренней (то есть не была сокращена), то $R_{\alpha_1}(t) > R_{\alpha_1}(T_t)$. □

Теорема 2.3 *Для любого $\beta > \alpha$ дерево $T(\beta)$ является поддеревом $T(\alpha)$ и получается из него в результате β-отсечений.*

Доказательство:
Давайте докажем по индукции, что дерево $(T_t)(\beta)$ является поддеревом $(T_t)(\alpha)$, что будет означать, что $T(\beta)$ является поддеревом $T(\alpha)$. Для любой внутренней вершин t, обе дочерних вершины которой листовые, это утверждение справедливо. Это вытекает из того факта, что если мы сокращаем эту вершину при α, то и при β мы ее сократим (обратное неверно).

Теперь рассмотрим внутреннюю вершину t. Имеем, что $(T_{t,\ell})(\beta) \subseteq (T_{t,\ell})(\alpha)$ и $(T_{t,r})(\beta) \subseteq (T_{t,r})(\alpha)$. В данном случае правое и левое поддеревья вершины t, полученные с использованием α, равны $(T_{t,r})(\alpha)$ и $(T_{t,\ell})(\alpha)$ соответственно (по теореме 2.1). Поддеревья с корнем в вершине t, полученные с использованием α и β, обозначим T_t^α и T_t^β соответственно.

Наша цель — показать, что если мы сейчас будем сокращать вершину t с использованием α, то и при использовании β мы ее сократим. То есть если $R_\alpha(t) \leqslant R_\alpha(T_t^\alpha)$ то и $R_\beta(t) \leqslant R_\beta(T_t^\beta)$. Это будет означать, что дерево $(T_t)(\beta)$ является поддеревом $(T_t)(\alpha)$.

Запишем:

$$R_\beta(t) = R_\alpha(t) + (\beta - \alpha) \leqslant R_\alpha(T_t^\alpha) + (\beta - \alpha) = R_\alpha\big((T_{t,\ell})(\alpha)\big) + R_\alpha\big((T_{t,r})(\alpha)\big) + (\beta - \alpha) \leqslant$$
$$\leqslant R_\alpha\big((T_{t,\ell})(\beta)\big) + R_\alpha\big((T_{t,r})(\beta)\big) + (\beta - \alpha) =$$
$$= R_\beta\big((T_{t,\ell})(\beta)\big) + R_\beta\big((T_{t,r})(\beta)\big) - (\beta - \alpha)(|T_t^\beta| - 1) =$$
$$= R_\beta(T_t^\beta) - (\beta - \alpha)(|T_t^\beta| - 1) \leqslant R_\beta(T_t^\beta).$$

Поскольку по теореме 2.1 дерево $T(\beta)$ минимизирует $R_\beta(T')$ по всем поддеревьям $T' \subseteq T$ с тем же корнем, и при этом оно является поддеревом $T(\alpha)$, то оно также минимизирует $R_\beta(T')$ по всем поддеревьям $T' \subseteq T(\alpha)$ с тем же корнем, значит его мы получим в результате β-отсечений дерева $T(\alpha)$.

□

Доказанные теоремы дают нам возможность сформулировать алгоритм построения описанных в начале последовательностей $T_K \subset \ldots, T_0$ и $\alpha_K > \cdots > \alpha_0$.

С помощью процедуры, описанной в теореме 2.2, найдем значение α_1 и дерево $T_1 = T_0(\alpha_1)$. Затем повторим для него процедуру, находим $\alpha_2 > \alpha_1$ (по теореме 2.2) и $T_2 = T_0(\alpha_2)$. Будем повторять это до тех пор, пока не получим тривиальное дерево T_K, состоящее лишь из корня дерева T_0. Теоремы 2.2 и 2.3 говорят, что для любого i дерево T_i является α-оптимальным поддеревом дерева T_0 для любого $\alpha \in [\alpha_i, \alpha_{i+1})$.

Мы получили следующую процедуру прунинга большого дерева T:

1. Положим $k = 0$, $T_0 = T$.
2. Положим $\alpha = \infty$.
3. Обходим внутренние вершины t дерева снизу вверх и вычисляем $R(T_t)$, $|T_t|$ и $g(t, T)$. Полагаем $\alpha = \min\bigl(\alpha, g(t, T)\bigr)$.
4. Обходим вершины сверху вниз и сокращаем вершины t, для которых $g(t, T) = \alpha$.
5. Полагаем $k = k + 1$, $\alpha_k = \alpha$, $T_k = T$.
6. Если T имеет больше одной вершины — возвращаемся к шагу 2.

2.2 Выбор параметра α

Как же выбрать одно из построенных таким образом деревьев T_i? Чаще всего используют два варианта. Один вариант — разбить обучающую выборку в начале перед наращиванием большого на две части и использовать вторую из частей исключительно для выбора дерева T_{i^*}, имеющего наименьшее значение функционала $R(T)$ на ней. Второй — с помощью скользящего контроля. Для каждого разбиения выборки на одной из частей строится последовательность деревьев $\{T_i\}$ и на второй части считаются значения функционала $R(T_i)$ для построенных деревьев. Найденные значения усредняются и α^* выбирается из интервала с наименьшим значением функционала.

Список литературы

[1] *Breiman, L., Friedman, J., Olshen, R. and Stone, C.* Classification and regression trees. Wadsworth, New York, 1984.

[2] *Ripley, B. D.* Pattern Recognition and Neural Networks. Cambridge University Press, 1996.