Школа анализа данных
Машинное обучение, часть 1
Теоретическое домашнее задание №1

Кузина Е.М.

18 марта 2020

Задача 1 Кроссвалидация, LOO, k-fold.
Объясните, стоит ли использовать оценку leave-one-out-CV или k-fold-CV с небольшим k в случае, когда:

- обучающая выборка содержит очень малое количество объектов;
- обучающая выборка содержит очень большое количество объектов.

Решение:
Распишем формулы для оценок leave-one-out-CV (LOO) и k-fold-CV (CV). Пусть рассматривается выборка объектов $X = \{x_1, \dots, x_L\}$. Тогда оценка leave-one-out-CV рассчитывается по следующей формуле:

$$LOO(a, X^L) = \frac{1}{L}\sum_{i=1}^{L} \mathcal{L}(a(X^L \backslash \{x_i\}, x_i))$$

Для подсчета оценки k-fold-CV нужно разбить выборку на k непересекающихся блоков одинаковой (или почти одинаковой) длины $l_1, \dots, l_k : X^L = X_1^{l_1} \sqcup X_2^{l_2} \sqcup \dots \sqcup X_k^{l_k}$, где $l_1 + l_2 + \dots + l_k = L$. Оценка k-fold-CV выглядит следующим образом:

$$CV(a, X^L) = \frac{1}{k}\sum_{i=1}^{k} \mathcal{L}(a(X^L \backslash X_i^{k_i}, X_i^{k_i}))$$

1. обучающая выборка содержит очень малое количество объектов

   В данном случае следует использовать оценку leave-one-out-CV. Так как размер выборки очень маленький, то можно быстро обучить модель L раз и подсчитать значение. Достоинтсвом этой оценки является то, что каждый объект выборки один раз участвует в контроле.

   С подсчетом k-fold-CV могут возникнуть проблемы, если k > L. Если k=L, то оценка превращается в оценку контроля по отдельным объектам (leave-one-out-CV).

2. обучающая выборка содержит очень большое количество объектов

   Так как выборка содержит очень большое количество объектов, то подсчёт оценки LOO достаточно ресурсоёмкий. Лучше всего в этом случае воспользоваться оценкой k-fold-CV, потому что приходится обучать модель небольшое количество k раз.

Ответ: Для выборки с малым количеством элементов лучше всего использовать оценку leave-one-out-CV, с большим количеством-k-fold-CV.

Задача 2. Логистическая регрессия, вывод функции потерь.

Рассмотрим выборку объектов $X = \{x_1, \ldots, x_l\}$ и их целевых меток $Y = \{y_1, \ldots, y_l\}$, где $y_i \in \{0, 1\}$. Предположим, что мы хотим обучить линейный классификатор:

$$Q(w, X^l) = \sum_{i=1}^{l} \mathcal{L}(y_i, \langle w, x_i \rangle) \to \min_w,$$

где w – веса линейной модели, $\mathcal{L}(y, z)$ – некоторая гладкая функция потерь.

Так как решается задача двухклассовой классификации, то будем обучать классификатор предсказывать вероятности принадлежности объекта классу 1, то есть решать задачу логистической регрессии. Для измерения качества такого классификатора обычно используют правдоподобие $P(Y|X)$ целевых меток Y при заданных объектах X в соответствии с предсказанными распределениями p — чем выше правдоподобие, тем точнее классификатор. Для удобства с вычислительной точки зрения обычно используется отрицательный логарифм правдоподобия, также называемый LogLoss (Logarithmic Loss). Будем считать, что пары объект-ответ (x_i, y_i) независимы между собой для разных i.

1. Покажите что:

$$\text{LogLoss} = -\text{LogLikelihood} = -\log(P(Y|X)) = -\sum_{i=1}^{l} (y_i \log \tilde{y}_i + (1 - y_i) \log(1 - \tilde{y}_i)).$$

Решение: Пусть алгоритм $\tilde{y}(x)$ предсказывает вероятность принадлежности объекта x классу 1. Тогда вероятность того, что объект x_i встретиться с классом $y_i \in \{0, 1\}$, равна $P(y_i|x_i) = \tilde{y}(x_i)^{y_i}(1 - \tilde{y}(x_i))^{1-y_i}$. Правдоподобие выбрки $P(Y|X) = \prod_{i=1}^{l} P(y_i|x_i) = \prod_{i=1}^{l} \tilde{y}(x_i)^{y_i}(1 - \tilde{y}(x_i))^{1-y_i}$. Следовательно,

$$\log(P(Y|X)) = \log(\prod_{i=1}^{l} \tilde{y}(x_i)^{y_i}(1 - \tilde{y}(x_i))^{1-y_i}) = \sum_{i=1}^{l} \log(\tilde{y}(x_i)^{y_i}(1 - \tilde{y}(x_i))^{1-y_i}) =$$

$$= \sum_{i=1}^{l} (\log \tilde{y}(x_i)^{y_i} + \log(1 - \tilde{y}(x_i))^{1-y_i}) = \sum_{i=1}^{l} (y_i \log \tilde{y}(x_i) + (1 - y_i) \log(1 - \tilde{y}(x_i))) =$$

$$= \sum_{i=1}^{l} (y_i \log \tilde{y}_i + (1 - y_i) \log(1 - \tilde{y}_i))$$

В итоге получаем:

$$\text{LogLoss} = -\text{LogLikelihood} = -\log(P(Y|X)) = -\sum_{i=1}^{l} (y_i \log \tilde{y}_i + (1 - y_i) \log(1 - \tilde{y}_i)).$$

2. Для того, чтобы классификатор возвращал числа из отрезка $[0, 1]$, положите

$$p(y_i = 1|x_i) = \sigma(\langle w, x_i \rangle) = \frac{1}{1 + \exp(-\langle w, x_i \rangle)};$$

сигмоидная функция монотонно возрастает, поэтому чем больше скалярное призведение, тем большая вероятность положительного класса будет предсказана объекту.

Подставьте трансформированный ответ линейной модели в логарифм правдоподобия. К какой функции потерь мы пришли? (Обратите внимание, что функция обычно записывается для классов $\{-1, 1\}$).

Решение: Перепишем логарифм правдоподобия (LogLoss) для классов $\{-1, 1\}$:

$$\text{LogLoss} = -\text{LogLikelihood} = -\log(P(Y|X)) = -\log(\prod_{i=1}^{l} \tilde{y}(x_i)^{[y_i=1]}(1-\tilde{y}(x_i))^{[y_i=-1]}) =$$

$$= -\sum_{i=1}^{l} \log(\tilde{y}(x_i)^{[y_i=1]}(1-\tilde{y}(x_i))^{[y_i=-1]}) = -\sum_{i=1}^{l} (\log \tilde{y}(x_i)^{[y_i=1]} + \log(1-\tilde{y}(x_i))^{[y_i=-1]}) =$$

$$= -\sum_{i=1}^{l} ([y_i = 1] \log \tilde{y}(x_i) + [y_i = -1] \log(1-\tilde{y}(x_i)))$$

$\tilde{y}(x_i)$ предсказывает вероятность принадлежности объекта x_i классу 1, подставив вместо него

$$p(y_i = 1|x_i) = \sigma(\langle w, x_i \rangle) = \frac{1}{1 + \exp(-\langle w, x_i \rangle)},$$

в формулу выше, получаем:

$$-\sum_{i=1}^{l} ([y_i = 1] \log \frac{1}{1 + \exp(-\langle w, x_i \rangle)} + [y_i = -1] \log \frac{\exp(-\langle w, x_i \rangle)}{1 + \exp(-\langle w, x_i \rangle)}) =$$

$$= -\sum_{i=1}^{l} ([y_i = 1] \log \frac{1}{1 + \exp(-\langle w, x_i \rangle)} + [y_i = -1] \log \frac{1}{1 + \exp(\langle w, x_i \rangle)}) =$$

$$= -\sum_{i=1}^{l} \log \frac{1}{1 + \exp(-y_i \langle w, x_i \rangle)} = \sum_{i=1}^{l} \log(1 + \exp(-y_i \langle w, x_i \rangle))$$

Ответ: Получаем логарифмическую функцию потерь

Задача 3. Логистическая регрессия, решение оптимизационной задачи.

1. Докажите, что в случае линейно разделимой выборки не существует вектора параметров (весов), который бы максимизировал правдоподобие вероятностной модели логистической регрессии в задаче двухклассовой классификации.

   Решение: Докажем методом от противного. Предположим, что существует вектор весов w, такой что функция правдоподобия максимальна, т.е.:

   $$p(x, y|w) = \prod_{i=1}^{l} p(x_i, y_i|w) \to \max_w$$

   Распишем функцию правдоподобия:

   $$p(x, y|w) = \prod_{i=1}^{l} p(x_i, y_i|w) = \prod_{i=1}^{l} P(y_i|x_i, w) p(x_i)$$

   Для случая линейно разделимой выборки $P(y_i|x_i, w) = 1 \; \forall i = 1, .., l$. Тогда $p(x, y|w) = \prod_{i=1}^{l} p(x_i)$.

   Полученное выражение не зависит от w, поэтому нет вектора w, который бы максимизировал правдоподобие. Получили противоречие. Следовательно, не существует вектора весов, который бы максимизировал правдоподобие.

2. Предложите, как можно модифицировать вероятностную модель, чтобы оптимум достигался.

   Решение: Для достижения оптимума нужно, чтобы выборка не была линейно разделимой. Чтобы этого добиться, можно, например, добавить шум в данные.

3. Выпишите формулы пересчета значений параметров при оптимизации методом градиентного спуска для обычной модели логистической регрессии и предложенной модификации.

   Решение:

   (a) для обычной логистической регрессии

   Пусть X^l - выборка: $(x_i, y_i)_{i=1}^l$, где x_i-объекты, y_i - класс, к которому принадлежит объект x_i. Введем функцию $\sigma(z) = \frac{1}{1+e^{-z}}$ - сигмоидная функция.

   Выпишем формулы для пересчета вектора весов w методом градиентного спуска для обычной модели логистической регрессии. Для этого введем функциоанал потерь:

$$Q(w) = \sum_{i=1}^{l} ln\left(1 + e^{-y_i\langle w, x_i\rangle}\right)$$

   Найдем $\nabla Q(w) = (\frac{\partial Q}{\partial w_j})_{j=1}^n$:

$$\frac{\partial Q}{\partial w_j} = \sum_{i=1}^{l} \frac{1}{1 + e^{-y_i\langle w, x_i\rangle}} * e^{-y_i\langle w, x_i\rangle} * (-y_i x_{ij}) = \sum_{i=1}^{l} \frac{1}{1 + e^{y_i\langle w, x_i\rangle}} * (-y_i x_{ij}) =$$

$$= -\sum_{i=1}^{l} \frac{y_i x_{ij}}{1 + e^{y_i\langle w, x_i\rangle}} = -\sum_{i=1}^{l} \sigma(-y_i\langle w, x_i\rangle) y_i x_{ij}$$

   Формулы для перерасчета:

   $w_j^0 :=$ начальное приближение

   $w_j^{t+1} := w_j^t - h\frac{\partial Q}{\partial w_j} = w_j^t + h\sum_{i=1}^{l} \sigma(-y_i\langle w, x_i\rangle) y_i x_{ij}$

   В матричной форме:

   $w^0 :=$ начальное приближение вектора весов

   $w^{t+1} := w^t + h y_i x_i \sigma(-y_i\langle w, x_i\rangle)$

   (b) для модификации

Задача 4. Мультиномиальная регрессия.

В случае многоклассовой классификации логистическую регрессию можно обобщить: пусть для каждого класса k есть свой вектор весов w_k. Тогда вероятность принадлежности классу k запишем следующим образом:

$$P(y = k|x, W) = \frac{e^{\langle w_k, x\rangle}}{\sum_{j=1}^{K} e^{\langle w_j, x\rangle}}$$

Тогда оптимизируемая функция примет вид:

$$\mathcal{L}_{sm}(W) = -\sum_{i=1}^{N}\sum_{k=1}^{K} [y_i = k] \ln P(y_i = k|x_i, W), \text{ где } [y_i = k] = \begin{cases} 1, y_i = k, \\ 0, \text{иначе} \end{cases}$$

Пусть количество классов $K = 2$. Для простоты положим, что выборка линейно неразделима.

1. Единственно ли решение задачи? Почему?

   Решение: В случае двух классов будем строить алгоритм $a(x) = sign(\langle w, x\rangle)$. Покажем, что при линейно зависимых признаках оптимизационная задача $Q(w) \to \min$ может иметь бесконечно много решений.

   Пусть признаки линейно зависимы, это по определению означает, что $\exists u : \forall$ объекта x $\langle u, x\rangle = 0$. Предположим, что w - оптимальный вектор весов, но тогда при векторе весов $w + cu$ (где c-константа из R) алгоритм будет давать точно такие же ответы, так как:

$$\langle w + cu, x\rangle = \langle w, x\rangle + \langle cu, x\rangle = \langle w, x\rangle$$

   Ответ: Решение задачи может быть не единственным.

2. Покажите, что предсказанные распределения вероятностей на классах в случае логистической и мультиномиальной регрессий будут совпадать.

   Решение: Распишем $\mathcal{L}_{sm}(W)$ для количества классов $K = 2$:

$$\mathcal{L}_{sm}(W) = -\sum_{i=1}^{N}([y_i = 1]\ln P(y_i = 1|x_i, W) + [y_i = 2]\ln P(y_i = 2|x_i, W)) =$$

$$= -\sum_{i=1}^{N}([y_i = 1]\ln\frac{e^{\langle w_1,x\rangle}}{e^{\langle w_1,x\rangle} + e^{\langle w_2,x\rangle}} + [y_i = 2]\ln\frac{e^{\langle w_2,x\rangle}}{e^{\langle w_1,x\rangle} + e^{\langle w_2,x\rangle}}) =$$

$$= -\sum_{i=1}^{N}([y_i = 1]\ln\frac{1}{1 + e^{\langle w_2-w_1,x\rangle}} + [y_i = 2]\ln\frac{1}{e^{\langle w_1-w_2,x\rangle} + 1})$$

$$= -\sum_{i=1}^{N}([y_i = 1]\ln\frac{1}{1 + e^{-\langle w_1-w_2,x\rangle}} + [y_i = 2]\ln\frac{1}{1 + e^{\langle w_1-w_2,x\rangle}})$$

   Если перенумеровать 2 класс в -1, то получаем следующее:

$$-\sum_{i=1}^{N}([y_i = 1]\ln\frac{1}{1 + e^{-\langle w_1-w_2,x\rangle}} + [y_i = -1]\ln\frac{1}{1 + e^{\langle w_1-w_2,x\rangle}}) =$$

$$= -\sum_{i=1}^{N}\ln\frac{1}{1 + e^{-y_i\langle w_1-w_2,x\rangle}} = \sum_{i=1}^{N}\ln(1 + e^{-y_i\langle w_1-w_2,x\rangle})$$

   Получаем точно такую же функцию потерь, что и в задаче 2 пункт 2 с тем отличием, что вектор весов w должен быть равен $w_1 - w_2$.

Задача 5 Решающие деревья, константное предсказание, функции потерь.

Допустим, при построении решающего дерева в некоторый лист попало N объектов $x_1, ..., x_N$ с метками $y_1, ..., y_N$. Предсказание в каждом листе дерева — константа. Найдите, какое значение $\tilde{y}$ должен предсказывать этот лист для минимизации следующих функций потерь:

1. Mean Squared Error (средний квадрат ошибки) для задачи регрессии:

$$Q = \frac{1}{N}\sum_{i=1}^{N}(y_i - \tilde{y})^2;$$

   Решение: Продифференцируем функцию потерь по аргументу $\tilde{y}$ и приравняем к 0:

$$\frac{\partial Q}{\partial \tilde{y}} = -\frac{2}{N}\sum_{i=1}^{N}(y_i - \tilde{y}) = 0$$

$$\tilde{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

Ответ: $\tilde{y} = \frac{1}{N}\sum\limits_{i=1}^{N} y_i$

2. Mean Absolute Error (средний модуль отклонения) для задачи регрессии:

$$Q = \frac{1}{N}\sum_{i=1}^{N}|y_i - \tilde{y}|.$$

Решение: Будем искать решение среди $y_i, i = 1, .., N$. Для этого рассмотрим вариационный ряд $\{y\}$: $y_{k1} <= y_{k2} <= ... <= y_{kN}$. Тогда MAE для $\tilde{y} = y_k$:

$$Q = \frac{1}{N}\sum_{i=1}^{k_N}|y_i - \tilde{y}| = \frac{1}{N}\sum_{i=1}^{k}(-y_i + y_k) + \frac{1}{N}\sum_{i=k+1}^{k_N}(y_i - y_k)$$

Теперь покажем, что сначала значение Q уменьшается, а затем увеличивается при переборе значений из вариационного ряда $\{y\}$. Для этого рассмотрим и сравним 2 функции потерь: при $\tilde{y} = y_k$ и $\tilde{y} = y_{k+1}$:

$$Q_k \vee Q_{k+1}$$

$$\frac{1}{N}\sum_{i=1}^{k}(-y_i + y_k) + \frac{1}{N}\sum_{i=k+1}^{k_N}(y_i - y_k) \vee \frac{1}{N}\sum_{i=1}^{k+1}(-y_i + y_{k+1}) + \frac{1}{N}\sum_{i=k+2}^{k_N}(y_i - y_{k+1})$$

$$\sum_{i=1}^{k}(-y_i + y_k) + \sum_{i=k+1}^{k_N}(y_i - y_k) \vee \sum_{i=1}^{k+1}(-y_i + y_{k+1}) + \sum_{i=k+2}^{k_N}(y_i - y_{k+1})$$

$$-\sum_{i=1}^{k} y_i + ky_k + \sum_{i=k+1}^{k_N} y_i - (k_N - k)y_k \vee -\sum_{i=1}^{k+1} y_i + (k+1)y_{k+1} + \sum_{i=k+2}^{k_N} y_i - (k_N - k - 1)y_{k+1}$$

$$ky_k + y_{k+1} - (k_N - k)y_k \vee -y_{k+1} + (k+1)y_{k+1} - (k_N - k - 1)y_{k+1}$$

$$(2k - k_N)y_k \vee (2k - k_N)y_{k+1}$$

$y_k \leq y_{k+1}$, поэтому знак неравенства будет определяться $2k - k_N$: если $k < \frac{k_N}{2}$, то $Q_k > Q_{k+1}$, и если $k > \frac{k_N}{2}$, то $Q_k < Q_{k+1}$. В итоге получаем, что MAE уменьшается при $\tilde{y} = y_i$, где $i = k_1, .., \frac{k_N}{2}$, и увеличивается при $\tilde{y} = y_i$, где $i = \frac{k_N}{2}, .., k_N$. Следовательно, MAE имеет минимум при $\tilde{y} = y_{\frac{k_N}{2}}$, что является медианой.

Ответ: $\tilde{y} = median(\{y_i\}_{i=1}^{N})$

3. LogLoss (логарифмические потери) для задачи классификации:

$$Q = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log \tilde{y} + (1-y_i)\log(1-\tilde{y})\right), \quad \tilde{y} \in [0,1], \quad y_i \in \{0,1\}.$$

Решение: Аналогично пункту 1:

$$\frac{\partial Q}{\partial \tilde{y}} = -\frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i}{\tilde{y}} - \frac{1-y_i}{1-\tilde{y}}\right) = -\frac{1}{N}\sum_{i=1}^{N}\frac{y_i - \tilde{y}}{\tilde{y}(1-\tilde{y})}$$

$\tilde{y} \in [0,1]$, поэтому знаменатель может обратиться в 0. Чтобы этого избежать, немного сузим отрезок следующим образом: $[\varepsilon, 1-\varepsilon]$, где ε - небольшое значение, предотвращающее обращение знаменателя в 0. Тогда, приравняв производную к 0 и вычислив $\tilde{y}$, получаем, что минимум логарифмических потерь достигается при $\tilde{y} = \frac{1}{N}\sum\limits_{i=1}^{N} y_i$

Ответ: $\tilde{y} = \frac{1}{N}\sum\limits_{i=1}^{N} y_i$

Задача 6 Решающие деревья, функции потерь, impurity functions.

$$\Phi(U) - \frac{|U_1|}{|U|}\Phi(U_1) - \frac{|U_2|}{|U|}\Phi(U_2) \to \max$$

таким выражением в лекции задается критерий, по которому происходит ветвление вершины решающего дерева. Давайте разберемся подробнее.

Impurity function $\Phi(U)$ («функция нечистоты» или «функция неопределенности») используется для того, чтобы измерить степень неоднородности целевых меток $y_1, \ldots, y_l$ для множества объектов U размера l. Например, при обучении решающего дерева в текущем листе выбирается такое разбиение множества объектов U на два непересекающихся множества U_1 и U_2, чтобы impurity function $\Phi(U)$ исходного множества U как можно сильнее превосходила нормированную impurity function в новых листьях $\frac{|U_1|}{|U|}\Phi(U_1) + \frac{|U_2|}{|U|}\Phi(U_2)$. Отсюда и получается, что нужно выбрать разбиение, решающее задачу

$$\Phi(U) - \frac{|U_1|}{|U|}\Phi(U_1) - \frac{|U_2|}{|U|}\Phi(U_2) \to \max.$$

Полученную разность называют Gain (выигрыш), и она показывает, на сколько удалось уменьшить «неопределенность» от разбиения листа два новых.

В соответствии с одним из возможных определений, impurity function — это значение функционала ошибки $Q = \frac{1}{l}\sum\limits_{i=1}^{l}\mathcal{L}(y_i, \tilde{y})$ в листе с множеством объектов U при константном предсказании $\tilde{y}$, оптимальном для Q (см. задачу 7):

$$\Phi(U) = \frac{1}{l}\sum_{i=1}^{l}\mathcal{L}(y_i, \tilde{y}).$$

Понятно, что каждому критерию разбиения соответствует своя impurity function $\Phi(U)$, а в основе каждой $\Phi(U)$ лежит некоторая функция потерь. Давайте разберемся, откуда берутся различные критерии разбиения.

1. Покажите, что для квадратичных потерь $\mathcal{L}(y_i, \tilde{y}) = (y_i - \tilde{y})^2$ в задаче регрессии $y_i \in \mathbb{R}$ impurity function $\Phi(U)$ равна выборочной дисперсии целевых меток объектов, попавших в лист дерева.

   Решение: Выборочная дисперсия целевых меток объектов, попавших в лист дерева равна:

$$\frac{1}{l}\sum_{i=1}^{l}(y_i - \frac{1}{l}\sum_{i=1}^{l} y_i)^2$$

   В соответствии с определением impurity function $\Phi(U)$ - это значение функционала $Q = \frac{1}{l}\sum\limits_{i=1}^{l}\mathcal{L}(y_i, \tilde{y})$ в листе с множеством объектов U при константном предсказании $\tilde{y}$, оптимальном для Q. Так как нужно минимизировать функционал ошибки Q для квадратичных потерь, то из задачи 5 пункта 1 известно, что это достигается при $\tilde{y} = \frac{1}{l}\sum\limits_{i=1}^{l} y_i$. Подставим это значение в $\Phi(U)$:

$$\Phi(U) = \frac{1}{l}\sum_{i=1}^{l}\mathcal{L}(y_i, \tilde{y}) = \frac{1}{l}\sum_{i=1}^{l}(y_i - \tilde{y})^2 = \frac{1}{l}\sum_{i=1}^{l}(y_i - \frac{1}{l}\sum_{i=1}^{l} y_i)^2$$

   Таким образом, $\Phi(U)$ равна выборочной дисперсии целевых лбъектов, попавших в лист дерева.

2. Покажите, что для функции потерь Logloss $\mathcal{L}(y_i, \tilde{y}) = -y_i \log(\tilde{y}) - (1 - y_i)\log(1 - \tilde{y})$ в задаче классификации $y_i \in \{0, 1\}$ impurity function $\Phi(U)$ соответствует энтропийному критерию разбиения.

   Решение:

$$\Phi(U) = \frac{1}{l}\sum_{i=1}^{l}\mathcal{L}(y_i, \tilde{y}) = -\frac{1}{l}\sum_{i=1}^{l}(y_i log(\tilde{y}) + (1-y_i) log(1-\tilde{y})) = -\frac{1}{l}\sum_{i=1}^{l} y_i log(\tilde{y}) - \frac{1}{l}\sum_{i=1}^{l}(1-y_i) log(1-\tilde{y}) =$$

$$= -\frac{1}{l}\sum_{i=1}^{l} y_i log(\tilde{y}) - (1 - \frac{1}{l}\sum_{i=1}^{l} y_i) log(1 - \tilde{y})$$

   В задаче 5 пункт 3 было показано, что при оптимальной функции потерь $\tilde{y} = \frac{1}{l}\sum\limits_{i=1}^{l} y_i$. Тогда $\Phi(U) = -\tilde{y} log(\tilde{y}) - (1 - \tilde{y}) log(1 - \tilde{y})$. Получаем энтропийный критерий разбиения. Что и требовалось показать.

Задача 7 Решающие деревья, индекс Джини.

Пусть имеется построенное решающее дерево для задачи многоклассовой классификации. Рассмотрим лист дерева с номером m и объекты R_m, попавшие в него. Обозначим за p_{mk} долю объектов k-го класса в листе m. *Индексом Джини* этого листа называется величина

$$\sum_{k=1}^{K} p_{mk}(1 - p_{mk}),$$

где K — общее количество классов. Индекс Джини обычно служит мерой того, насколько хорошо в данном листе выделен какой-то один класс (см. impurity function в предыдущей задаче).

1. Поставим в соответствие листу m алгоритм классификации $a(x)$, который предсказывает класс случайно, причем класс k выбирается с вероятностью p_{mk}. Покажите, что матожидание частоты ошибок этого алгоритма на объектах из R_m равно индексу Джини.

   Решение: Пусть ν - частота ошибок этого алгоритма на объектах из R_m. Найдем её матожидание:

$$E\nu = E\frac{\sum\limits_{x_i \in R_m} [y_i \neq a(x_i)]}{N_m} = \frac{1}{N_m} \sum_{x_i \in R_m} E[y_i \neq a(x_i)] = \frac{1}{N_m} \sum_{x_i \in R_m} (1 - p_{m,yi}) =$$

$$= \frac{1}{N_m} \sum_{k=1}^{K} \sum_{x_i \in R_m} [y_i = k](1 - p_{mk}) = \sum_{k=1}^{K} p_{mk}(1 - p_{mk})$$

2. *Дисперсией класса k* назовем дисперсию выборки $\{[y_i = k] : \ x_i \in R_m\}$, где y_i — класс объекта x_i, $[f]$ — индикатор истинности выражения f, равный 1 если f верно, и нулю в противном случае, а R_m — множество объектов в листе. Покажите, что сумма дисперсий всех классов в заданном листе равна его индексу Джини.

   Решение:

   Рассмотри случайную величину $\varepsilon = \begin{cases} 1, \text{класс k} \\ 0, \text{иначе} \end{cases}$

   Матожидание этой случайной величины равняется $E\varepsilon = 1 * p_{mk} + 0 * (1 - p_{mk}) = p_{mk}$, а дисперсия $D\varepsilon = (1 - p_{mk})^2 p_{mk} + (0 - p_{mk})^2(1 - p_{mk}) = p_{mk}(1 - p_{mk})$

   Следовательно, сумма дисперсий для всех классов будет равняться $\sum\limits_{k=1}^{K} p_{mk}(1 - p_{mk})$.

Задача 8 Бинарные решающие деревья, MSE.

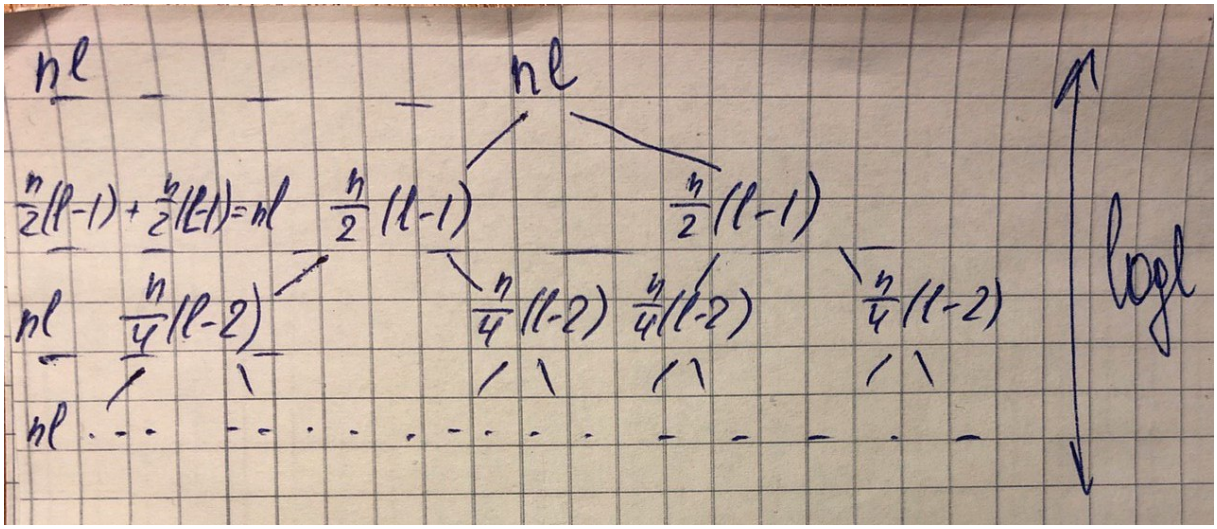
Предложите алгоритм построения **оптимального** бинарного решающего дерева для задачи регрессии на l объектах в n-мерном пространстве с асимптотической сложностью $O(nl \log l)$. В качестве предикатов нужно рассматривать пороговые правила (наиболее распространенный случай на практике). Для простоты можно считать, что получающееся дерево близко к сбалансированному (т.е. его глубина имеет порядок $O(\log l)$) и в качестве функции ошибки используется Mean Squared Error (MSE):

$$Q = \frac{1}{l} \sum_{i=1}^{l} (y_i - \tilde{y}_i)^2.$$

Под оптимальностью в данной задаче подразумевается, что в каждом узле дерева делается оптимальное с точки зрения MSE разбиение на два поддерева.

Решение: Данный алгоритм можно реализовать рекурсивно. Будем разбивать выборку по пороговым правилам и находить минимальное значение MAE. Для каждого правила попытаемся разбить выборку на две части и минимизировать $MAE_{left} + MAE_{right}$ (для левой и правой подвыборки). После этого разбиваем выборку на две части для того порогового правила, где значение MAE минимально и запускаем алгоритм рекурсивно на каждой из двух получившихся подвыборках.

Асимптотика алгоритма: дерево рекурсии данного алгоритма выглядит следующим образом:



1. изначально у нас есть вся выборка размером $n * l$, поиск правила, которое удовлетворяет требованию, и разбиение выборки будет занимать $O(nl)$

2. после разделения выборки пополам (так как по условию задачи предполагаем, что получается дерево близкое к сбалансированному) и выбора правила (получаем, что на данном этапе уже есть $l-1$ правило), получаем асимптотику $O(\frac{n}{2}(l-1))$ для каждой из подзадач. В сумме на этом уровне асимптотика равна $O(nl)$

3. Можно продолжать показывать, что на каждом из следующих уровней асимптотика равна $O(nl)$. Всего же уровней $O(log(l))$. Тогда получаем, что наш алгоритм работает за время $O(nl\log(l))$

Задача 9 Метрические методы, kNN, устойчивость к шуму.

Известно, что метод ближайших соседей неустойчив к шуму. Рассмотрим модельную задачу бинарной классификации с одним признаком и двумя объектами обучающей выборки: $x_1 = 0.1$, $x_2 = 0.5$. Первый объект относится к первому классу, второй — ко второму. Добавим к объектам новый шумовой признак, распределенный равномерно на отрезке $[0, 1]$. Теперь каждый объект описывается уже двумя признаками. Пусть требуется классифицировать новый объект $u = (0, 0)$ в этом пространстве методом одного ближайшего соседа с евклидовой метрикой. Какова вероятность того, что после добавления шума второй объект окажется ближе к объекту u, чем первый?

Решение: Рассмотрим евклидово расстояние между объектами $a, b : \rho(a, b) = (a_1 - b_1)^2 + (a_2 - b_2)^2$ и учтем, что шумовые значения $\xi \in [0, 1]$, так как они распределены равномерно на отрезке [0,1]. Тогда искомая вероятность вычисляется следующим образом:

$$P(\rho(x_2, u) < \rho(x_1, u)) = P(x_2^2 + {\xi_2}^2 < x_1^2 + {\xi_1}^2) = P({\xi_2}^2 < {\xi_1}^2 - 0.24) = \int\limits_0^{\sqrt{0.76}} \left(\int\limits_{\sqrt{x_2^2+0.24}}^{1} dx_1 \right) dx_2 =$$

$$= \int\limits_0^{\sqrt{0.76}} (1 - \sqrt{x_2^2 + 0.24}) dx_2 = x_2 \Big|_0^{\sqrt{0.76}} - \frac{x_2}{2}\sqrt{0.24 + x^2}\Big|_0^{\sqrt{0.76}} - 0.12 * ln\left(x_2 + \sqrt{0.24 + x_2^2}\right)\Big|_0^{\sqrt{0.76}} \approx 0.275$$

Вычисление интеграла приведено ниже:

$$\int_0^{\sqrt{0.76}} (1 - \sqrt{x^2 + 0.24})dx = \int_0^{\sqrt{0.76}} dx - \int_0^{\sqrt{0.76}} \sqrt{x^2 + 0.24}dx$$

1. $\int_0^{\sqrt{0.76}} dx = x\Big|_0^{\sqrt{0.76}}$

2. Чтобы посчитать $\int_0^{\sqrt{0.76}} \sqrt{x^2 + 0.24}dx$, сделаем замену переменной: $x = \sqrt{0.24}sh(t)$, $dx = \sqrt{0.24}ch(t)dt$:

$$\int_0^{\sqrt{0.76}} \sqrt{x^2 + 0.24}dx = 0.24 \int_0^{\sqrt{0.76}} ch^2(t)dt = \frac{0.24}{4} \int_0^{\sqrt{0.76}} (e^t + e^{-t})^2 dt = \frac{0.24}{4} \int_0^{\sqrt{0.76}} (e^{2t} + 2 - e^{-2t})dt =$$

$$= \frac{0.24}{4}(\frac{e^{2t}}{2} + 2t - \frac{e^{-2t}}{2})\Big|_0^{\sqrt{0.76}} = \frac{0.24}{4}(sh(2t) + 2t)\Big|_0^{\sqrt{0.76}} \quad (1)$$

Из замены получаем:

$$\frac{x}{\sqrt{0.24}} = sh(t)$$

$$\frac{x}{\sqrt{0.24}} = \frac{e^t - e^{-t}}{2}$$

$$e^{2t} - \frac{2x}{\sqrt{0.24}}e^t - 1 = 0$$

$$e^t = \frac{x}{\sqrt{0.24}} \pm \sqrt{\left(\frac{x}{\sqrt{0.24}}\right)^2 + 1}$$

Решение $e^t = \frac{x}{\sqrt{0.24}} - \sqrt{\left(\frac{x}{\sqrt{0.24}}\right)^2 + 1}$ не подходит, т.к не имеет решений (правая часть меньше 0). Получаем, что:

$$t = ln\left(\frac{x}{\sqrt{0.24}} + \sqrt{\left(\frac{x}{\sqrt{0.24}}\right)^2 + 1}\right) \quad (2)$$

$$sh(2t) = 2sh(t)ch(t) = 2sh(t)\sqrt{1 + sh^2(t)} = \frac{2x}{\sqrt{0.24}}\sqrt{1 + \left(\frac{x}{\sqrt{0.24}}\right)^2} \quad (3)$$

Подставим (2) и (3) в (1):

$$\frac{0.24}{4}\left(\frac{2x}{\sqrt{0.24}}\sqrt{1 + \left(\frac{x}{\sqrt{0.24}}\right)^2} + 2ln\left(\frac{x}{\sqrt{0.24}} + \sqrt{\left(\frac{x}{\sqrt{0.24}}\right)^2 + 1}\right)\right)\Big|_0^{\sqrt{0.76}} = \frac{x}{2}\sqrt{0.24 + x^2}\Big|_0^{\sqrt{0.76}} +$$

$$+0.12 * ln\left(x + \sqrt{0.24 + x^2}\right)\Big|_0^{\sqrt{0.76}}$$

Ответ: ≈ 0.275