Задачи прогнозирования, обобщающая способность, байесовский классификатор, скользящий контроль

Лектор – *Сенько Олег Валентинович*

Курс «Математические методы обучения»

Содержание лекции

Задачи прогнозирования

Задачи диагностики и прогнозирования некоторой величины Y по доступным значениям переменных $X_1, \ldots, X_n$ часто возникают в различных областях человеческой деятельности:

- постановка медицинского диагноза или результатов лечения по совокупности клинических и лабораторных показателей;
- прогноз свойств ещё не синтезированного химического соединения по его молекулярной формул;
- диагностика хода технологического процесса;
- диагностика состояния технического оборудования;
- прогноз финансовых индикаторов;
- и многие другие задачи

Модели прогнозирования

Для решения подобных задач могут быть использованы методы, основанные на использовании точных знаний. Например, могут использоваться методы математического моделирования, основанные на использовании физических законов. Однако сложность точных математических моделей нередко оказывается слишком высокой. Кроме того при использовании физических моделей часто требуется знание различных параметров, характеризующих рассматриваемое явление или процесс. Значения некоторых из таких параметров часто известны только приблизительно или неизвестны вообще. Все эти обстоятельства ограничивают возможности эффективного использования физических моделей.

Прогнозирования по прецендентам

В прикладных исследованиях нередко возникают ситуации, когда математическое моделирование, основанное на использовании точных законов оказывается затруднительны, но в распоряжении исследователей оказывается выборка прецедентов - результатов наблюдений исследуемого процесса или явления, включающих значения прогнозируемой величины Y и переменных $X_1, \ldots, X_n$. В этих случаях для решения задач диагностики и прогнозирования могут быть использованы методы, основанные на обучении по прецедентам.

Прогнозирование по прецедентам. Генеральная совокупность

Предположим, что задача прогнозирования решается для некоторого процесса или явления $\mathbf{F}$. Множество объектов, которые потенциально могут возникать в рамках $\mathbf{F}$, называется генеральной совокупностью, далее обозначаемой Ω . Предполагается, что прогнозируемая величина Y и переменные $X_1, \ldots, X_n$ заданы на Ω. Однако значение Y для некоторых объектов Ω может по разным причинам оказаться недоступным исследователю. При этом значения по крайней мере части переменных $X_1, \ldots, X_n$ известны. Целью математических методов прогнозирования, рассматриваемых в курсе, является построение алгоритма, вычисляющего недоступные значений Y по известным значениям переменных $X_1, \ldots, X_n$. Обычно генеральная совокупность рассматривается как множество элементарных событий, на котором заданы - алгебра событий Σ и вероятностная мера $\mathbf{P}$. То есть генеральная совокупность рассматривается как вероятностное пространство $\langle \Omega, \Sigma, P \rangle$.

Поиск алгоритма, вычисляющего осуществляется по выборке прецедентов, которая обычно является случайной выборкой объектов из Ω с известными значениями $Y, X_1, \ldots, X_n$, Выборку прецедентов также принято называть обучающей выборкой.
Обучающая выборка имеет вид $\tilde{S}_t = \{(y_1, \boldsymbol{x}_1), \ldots, (y_m, \boldsymbol{x}_m)\}$, где
y_j – значение переменной Y для объекта s_j, $j = 1, \ldots, m$;
$\boldsymbol{x}_j$ – значение вектора переменных $X_1, \ldots, X_n$ для объекта s_j;
m – число объектов в $\tilde{S}_t$.

Обучающая выборка

Обычно предполагается, что обучающая выборка может рассматриваться как независимая выборка объектов из Ω. Иными словами предполагается, что $\tilde{S}_t$ является элементом декартова произведения $\Omega_m = \Omega \times \ldots \times \Omega$. При этом предполагается, что на Ω_m задана σ-алгебра Σ_m, содержащая всевозможные декартовы произведения вида $\boldsymbol{a}_1 \times \ldots \times \boldsymbol{a}_m$, где $\boldsymbol{a}_i \in \Sigma$, $i = 1, \ldots, m$, и вероятностная мера P^m, удовлетворяющая условию

$$P^m(\boldsymbol{a}_1 \times \ldots \times \boldsymbol{a}_m) = \prod_{i=1}^{m} P(\boldsymbol{a}_i).$$

Методы, основанные на обучении по прецедентам

В процессе обучения производится поиск эмпирических закономерностей, связывающих прогнозируемую переменную Y с переменными $X_1, \ldots, X_n$. Данные закономерности далее используются при прогнозировании. Методы, основанные на обучении по прецедентам, также принято называть методами машинного обучения (machine learning).

Типы прогнозируемых величин

Прогнозируемая величина Y может иметь различную природу:

- принимать значения из отрезка непрерывной оси;
- принимать значения из конечного множества;
- являться кривой, описывающей вероятность возникновения некоторого критического события до различных моментов времени.

Задачи, в которых прогнозируемая величина принимает значения из множества, содержащего несколько элементов, принято называть задачей распознавания. Например, к задачам распознавания относятся задачи прогнозирования категориальных переменных.

Примеры задач машинного обучения

Задача распознавания (классификации) ириса на три класса. Здесь целевая переменная $Y \in \{setosa, versicolor, virginica\}$, признаки $X_1, \ldots, X_4 \in \mathbb{R}$.

Классы:

Setosa

Versicolor

Virginica

Признаки:

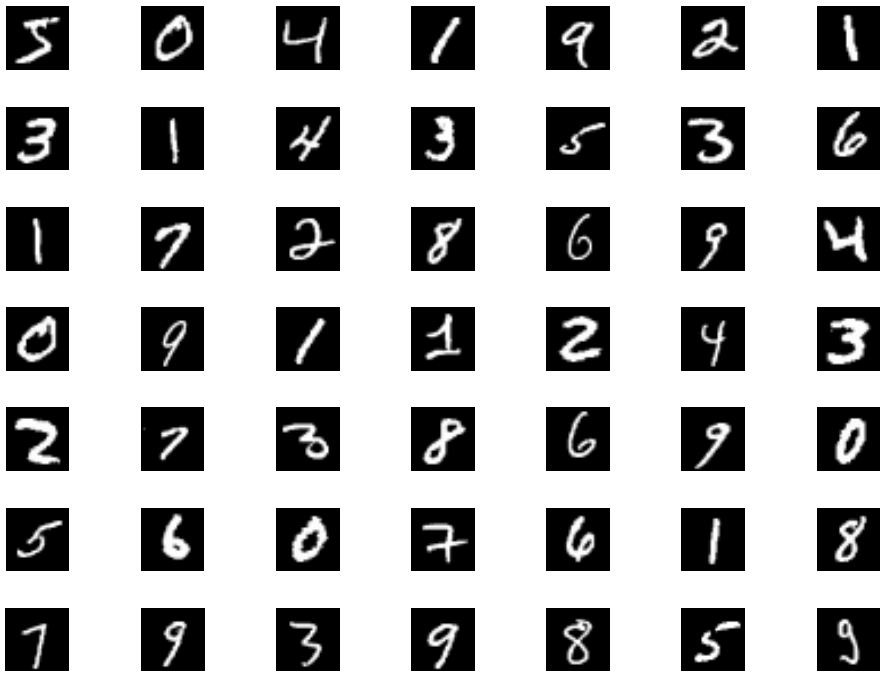
- длина чашелистика (см)
- ширина чашелистика (см)
- длина лепестка (см)
- ширина лепестка (см)

Данные: `http://archive.ics.uci.edu/ml/datasets/Iris`

Примеры задач машинного обучения

Задача распознавания рукописных цифр. Целевая переменная $Y \in \{0, 1, \ldots, 9\}$, признаки $X_1, X_2, \ldots, X_{784} \in [0, 255]$ – пикселы изображения размера 28×28.

Примеры объектов:



Данные: `http://yann.lecun.com/exdb/mnist/`

Примеры задач машинного обучения

Задача прогноза стоимости жилья в различных пригородах Бостона (задача восстановления регрессии).

Целевая переменная Y – цена жилья. Признаки:

- уровень криминала в районе
- концентрация окисей азота
- доля жилья, построенного до 1940 года
- среднее расстояние до основных районов концентрации рабочих мест
- уровень налогообложения
- отношение числа учителей к числу учеников в школах
- и другие

Данные: `http://archive.ics.uci.edu/ml/datasets/Housing`

Способы поиска закономерностей

Основным способом поиска закономерностей является поиск в некотором априори заданном семействе алгоритмов прогнозирования $\tilde{M} = \{A : \tilde{X} \to \tilde{Y}\}$ алгоритма, наилучшим образом аппроксимирующего связь переменных из набора $X_1, \ldots, X_n$ с переменной Y на обучающей выборке, где

$\tilde{X}$ – область возможных значений векторов переменных $X_1, \ldots, X_n$;
$\tilde{Y}$ – область возможных значений переменной Y.

Пусть $\lambda[y_j, A(\boldsymbol{x}_j)]$ – величина «потерь», произошедших в результате использования $A(\boldsymbol{x}_j)$ в качестве прогноза значения Y. Тогда одним из способов обучения является минимизация функционала эмпирического риска на обучающей выборке:

$$Q(\tilde{S}_t, A) = \frac{1}{m} \sum_{j=1}^{m} \lambda[y_j, A(\boldsymbol{x}_j)] \to \min_{A \in \tilde{M}}.$$

Частные случаи функции потерь

При прогнозировании непрерывных величин могут использоваться

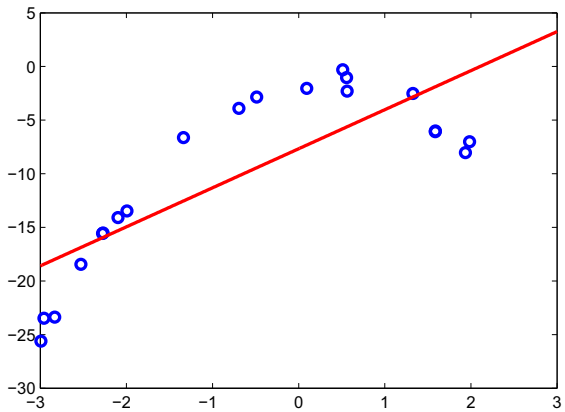
$\lambda[y_j, A(\boldsymbol{x}_j)] = (y_j - A(\boldsymbol{x}_j))^2$ – квадрат ошибки,
$\lambda[y_j, A(\boldsymbol{x}_j)] = |y_j - A(\boldsymbol{x}_j)|$ – модуль ошибки.

В случае задачи распознавания функция потерь может быть равной 0 при правильной классификации и 1 при ошибочной. При этом функционал эмпирического риска равен числу ошибочных классификаций.

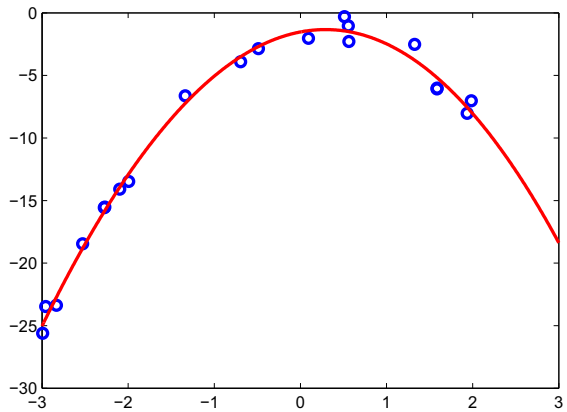
Примеры поиска закономерностей

Рассмотрим задачу восстановления регрессии по одному признаку. Здесь $\tilde{Y} = \mathbb{R}$, $\tilde{X} = \mathbb{R}$. Поиск зависимости между регрессионной переменной Y и признаком X в рамках семейства отображений $\tilde{M}$ осуществляется с помощью минимизации функционала эмпирического риска с функцией потерь $\lambda[y, A(x)] = (y - A(x))^2$ (т.н. метод наименьших квадратов).

Поиск зависимости в семействе линейных функций $\tilde{M} = \{y = ax + b, a, b \in \mathbb{R}\}$:



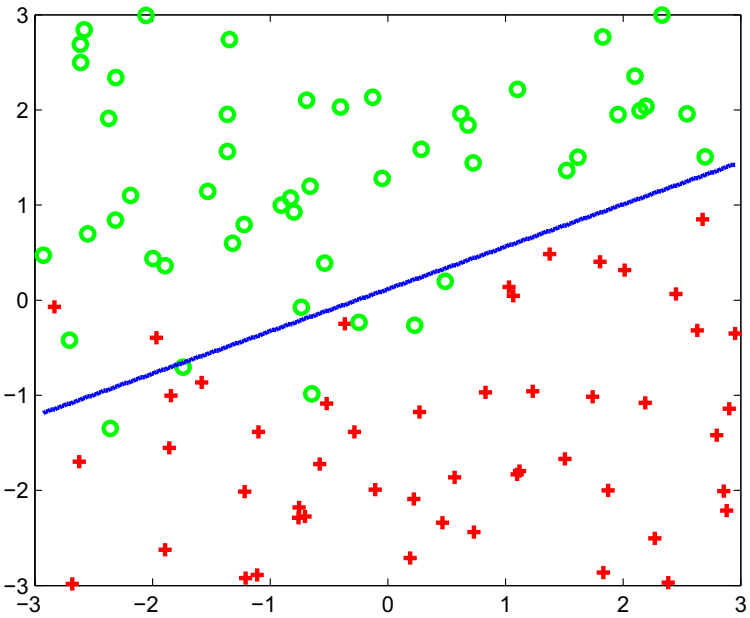
Поиск зависимости в семействе кубических функций $\tilde{M} = \{y = ax^3 + bx^2 + cx + d, a, b, c, d \in \mathbb{R}\}$:



Примеры поиска закономерностей

Рассмотрим задачу классификации на два класса по двум признакам.
Здесь $\tilde{Y} = \{1, 2\}$, $\tilde{X} = \mathbb{R}^2$.
Поиск зависимости в семействе линейных разделителей:

$$y = \begin{cases} 1, & \text{если } ax_1 + bx_2 + c \geq 0, \\ 2, & \text{иначе.} \end{cases}$$



Обобщающая способность

Точность алгоритма прогнозирования на всевозможных новых не использованных для обучения объектах, которые возникают в результате процесса, соответствующего рассматриваемой задаче прогнозирования, принято называть обобщающей способностью. Иными словами обобщающую способность алгоритма прогнозирования можно определить как точность на всей генеральной совокупности. Мерой обобщающей способности служит математическое ожидание потерь по генеральной совокупности $E_{\Omega}\{\lambda[Y, A(\boldsymbol{x})]\}$.

Обобщающая способность

Обобщающая способность может быть записана в виде

$$E_\Omega\{\lambda[Y, A(\boldsymbol{x})]\} = \int_M E\{\lambda[Y, A(\boldsymbol{x})]|\boldsymbol{x}\}p(\boldsymbol{x})dx_1 \dots dx_n,$$

где $p(\boldsymbol{x})$ – плотность вероятности в точке $\boldsymbol{x}$.
Интегрирование ведётся по области M, принадлежащей пространству $\mathbb{R}^n$ вещественных векторов размерности n, из которой принимают значения $X_1, \dots, X_n$.

Обобщающая способность

При решении задач прогнозирования основной целью является достижение максимальной обобщающей способности.

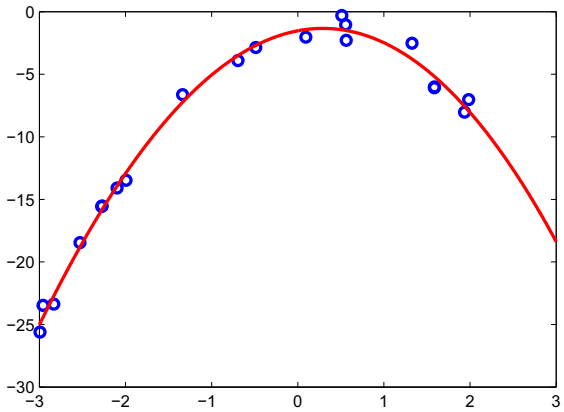
Эффект переобучения

Расширение модели $\tilde{M} = \{A : \tilde{X} \to \tilde{Y}\}$, увеличение её сложности, всегда приводит к повышению точности аппроксимации на обучающей выборке. Однако повышение точности на обучающей выборке, связанное с увеличением сложности модели, часто не ведёт к увеличению обобщающей способности. Более того, обобщающая способность может даже снижаться. Различие между точностью на обучающей выборке и обобщающей способностью при этом возрастает. Данный эффект называется эффектом переобучения.

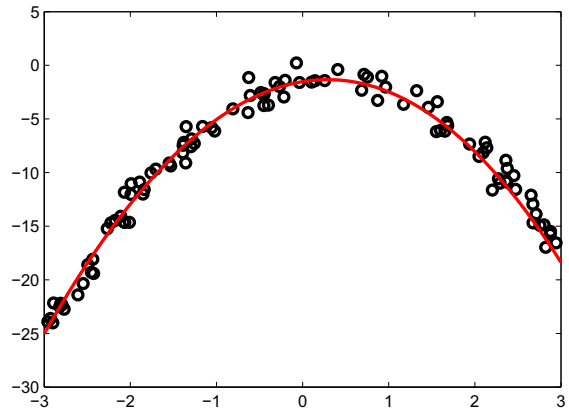
Примеры эффекта переобучения

Задача восстановления регрессии по одному признаку.

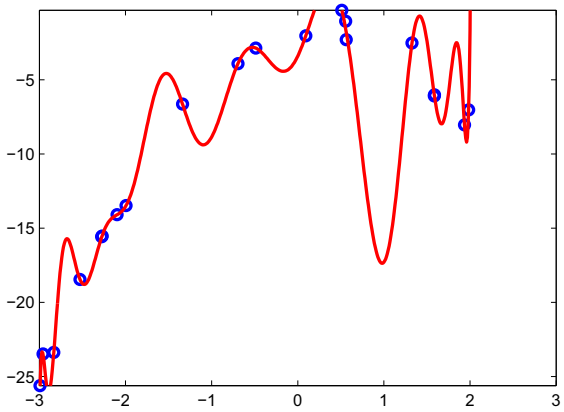
Восстановление кубической зависимости по обучающим данным:



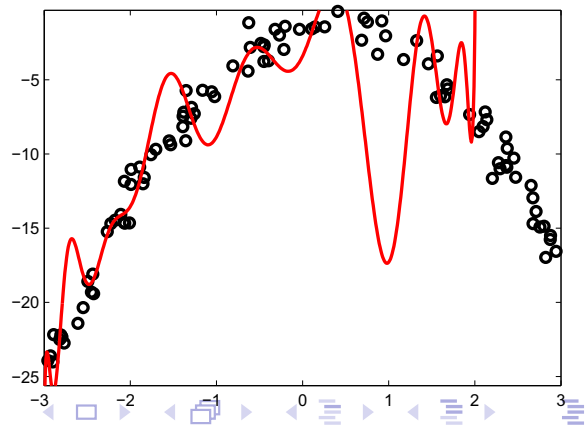
Применение восстановленной зависимости к тестовым данным из той же генеральной совокупности:



Увеличение сложности восстанавливаемой зависимости (степень полинома = 20) приводит к повышению точности на обучающих данных:



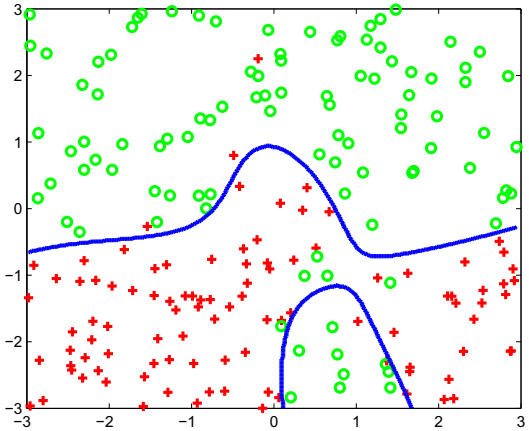
Применение сложной зависимости к тестовым данным обнаруживает переобучение:



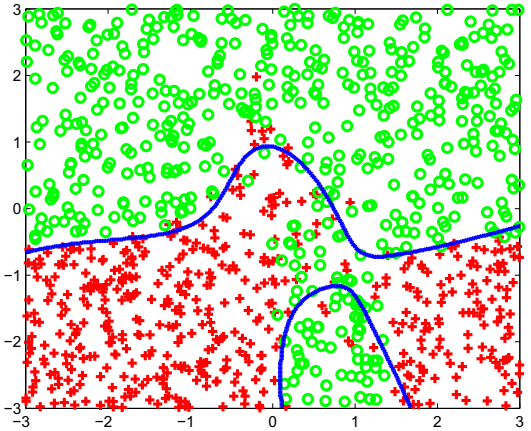
Примеры эффекта переобучения

Задача классификации на два класса по двум признакам.

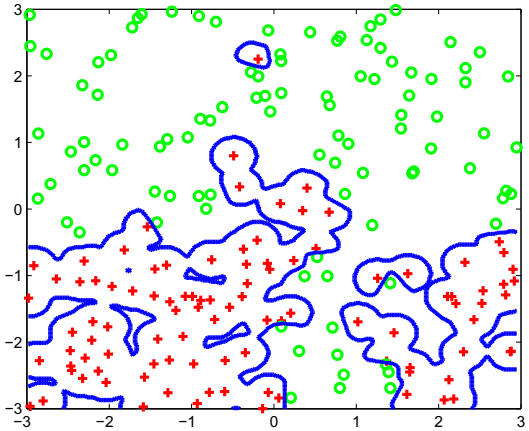
Поиск простой разделяющей кривой по обучающим данным (ошибка 5%):



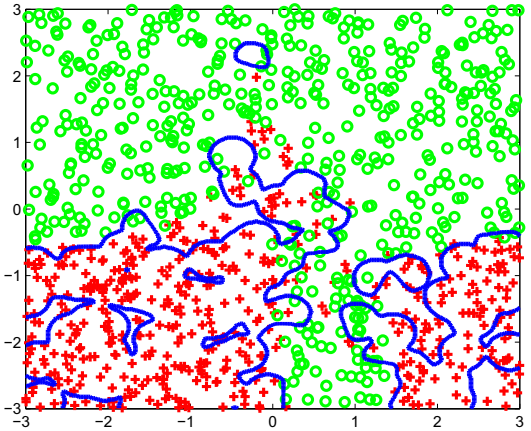
Применение разделяющей кривой к тестовым данным из той же генеральной совокупности (ошибка 6%):



Увеличение сложности разделяющей кривой приводит к 100-процентной точности на обучающих данных:



Применение сложной разделяющей кривой к тестовым данным обнаруживает переобучение (ошибка 14%):



Для какого алгоритма прогнозирования достигается максимальная обобщающая способность?

В случае, если при прогнозе Y в точке $\boldsymbol{x}$ используется величина $A(\boldsymbol{x})$, а величиной потерь является квадрат ошибки (т.е. $\lambda[y_j, A(\boldsymbol{x}_j)] = (y_j - A(\boldsymbol{x}_j))^2$), справедливо разложение:

$$E\{\lambda[Y, A(\boldsymbol{x})]|\boldsymbol{x}\} = E\{[Y - A(\boldsymbol{x})]^2|\boldsymbol{x}\} =$$
$$E\{[Y - E(Y|\boldsymbol{x}) + E(Y|\boldsymbol{x}) - A(\boldsymbol{x})]^2|\boldsymbol{x}\} =$$
$$E\{[Y - E(Y|\boldsymbol{x})]^2|\boldsymbol{x}\} + E\{[A(\boldsymbol{x}) - E(Y|\boldsymbol{x})]^2|\boldsymbol{x}\} +$$
$$2[A - E(Y|\boldsymbol{x})]E\{[Y - E(Y|\boldsymbol{x})]|\boldsymbol{x}\}.$$

Для какого алгоритма прогнозирования достигается максимальная обобщающая способность?

Здесь мы воспользовались простейшими свойствами условных математических ожиданий. Для произвольных случайных функций ζ_1 и ζ_2

$$E[(\zeta_1 + \zeta_2)|\boldsymbol{x}] = E[\zeta_1|\boldsymbol{x}] + E[\zeta_2|\boldsymbol{x}].$$

Для произвольной константы C и произвольной случайной функции ζ

$$E[C\zeta|\boldsymbol{x}] = CE[\zeta|\boldsymbol{x}].$$

Также $E[1|\boldsymbol{x}] = 1$.

Для какого алгоритма прогнозирования достигается максимальная обобщающая способность?

Однако

$$2[E(Y|\boldsymbol{x}) - A(\boldsymbol{x})]E\{[Y - E(Y|\boldsymbol{x})]|\boldsymbol{x}\} = 0.$$

Отсюда следует, что

$$E\{[Y - A(\boldsymbol{x})]^2|\boldsymbol{x}\} = [E(Y|\boldsymbol{x}) - A(\boldsymbol{x})]^2 + E\{[Y - E(Y|\boldsymbol{x})]^2|\boldsymbol{x}\}.$$

Из этой формулы хорошо видно, что наилучший прогноз должен обеспечивать алгоритм, вычисляющий прогноз как $A(\boldsymbol{x}) = E(Y|\boldsymbol{x})$.

Байесовский классификатор

Покажем, что при справедливости предположения о том, что всю доступную информацию о распределении объектов по классам содержат переменные $X_1, \ldots, X_n$, байесовский классификатор обеспечивает наименьшую ошибку распознавания. Пусть используется классификатор, относящий в некоторой точке $\boldsymbol{x}$ в классы $K_1, \ldots, K_L$ доли объектов $\nu_1(\boldsymbol{x}), \ldots, \nu_L(\boldsymbol{x})$, соответственно. Из предположении о содержаннии всей информации о классах переменными $X_1, \ldots, X_n$ следует, что внутри множества объектов с фиксированным $\boldsymbol{x}$ истинный номер класса не зависит от вычисленного прогноза. Откуда следует, что вероятность отнесения в класс K_i объекта, который классу K_i в действительности принадлежит составляет $nu(\boldsymbol{x})P(K_i|\boldsymbol{x})$. Вероятность ошибочного отнесения в классы отличные от K_i объекта, в действительности принадлежа- щего K_i , составляет $[1 - nu(\boldsymbol{x})]P(K_i|\boldsymbol{x})$.

Байесовский классификатор

Общая вероятность ошибочных классификаций в точке $\boldsymbol{x}$ составляет

$$\sum_{i=1}^{L}[1-\nu_i(\boldsymbol{x})]P(K_i|\boldsymbol{x}) = 1 - \sum_{i=1}^{L}\nu_i(\boldsymbol{x})P(K_i|\boldsymbol{x}).$$

Задача поиска минимума ошибки сводится к задаче линейного программирования вида

$$\sum_{i=1}^{L}\nu_i P(K_i|\boldsymbol{x}) \to \max_{\nu_1,\ldots,\nu_L},$$

$$\sum_{i=1}^{L}\nu_i = 1,$$

$$\nu_i \geq 0, i = 1,\ldots,L.$$

Байесовский классификатор

Одна из вершин симплекса, задаваемого ограничениями задачи линейного программирования, является её решением. Данное решение представляет собой бинарный вектор размерности L, имеющий вид (0, ..., 1, ..., 0). При этом значение 1 находится в позиции i', для которой выполняется набор неравенств $P(K_{i'}|\boldsymbol{x}) \geq P(K_i|\boldsymbol{x})$, $i = 1, \ldots, L$. В случае, если максимальная условная вероятность достигается только для одного класса $K_{i'}$, решение задачи линейного программирования достигается в единственной точке, задаваемой бинарным вектором с единственной 1, находящейся в позиции i'.

Байесовский классификатор

Предположим, что максимальная условная вероятность достигается для нескольких классов $K_{j(1)}, \ldots, K_{j(l')}$. Тогда решением задачи является вектор $\nu_1, \ldots, \nu_L$, компоненты которого удовлетворяют условиям:

$$\nu_i = 0, i \neq j(r), r = 1, \ldots, l'.$$

Из этого следует, что любая стратегия, при которой объекты относятся в один из классов $K_{j(1)}, \ldots, K_{j(l')}$, является оптимальной.

Однако для вычисления условных математических ожиданий $E(Y|\boldsymbol{x})$ или условных вероятностей $P(K_i|\boldsymbol{x})$, $i = 1, \ldots, L$, необходимы знания конкретного вида вероятностных распределений, присущих решаемой задаче. Такие знания в принципе могут быть получены с использованием известного метода максимального правдоподобия.

Метод максимального правдоподобия

Метод максимального правдоподобия (ММП) используется в математической статистике для аппроксимации вероятностных распределений по выборкам данных. В общем случае ММП требует априорных предположений о типе распределений. Значения параметров $(\theta_1, \ldots, \theta_r)$, задающих конкретный вид распределений, ищутся путём максимизации функционала правдоподобия. Функционал правдоподобия представляет собой произведение плотностей вероятностей на объектах обучающей выборки.

Метод максимального правдоподобия

Функционал правдоподобия имеет вид

$$L(\tilde{S}_t, \theta_1, \ldots, \theta_r) = \prod_{j=1}^{m} p(y_j, \boldsymbol{x}_j, \theta_1, \ldots, \theta_r).$$

Наряду с методом минимизации эмпирического риска метод ММП является одним из важнейших инструментов настройки алгоритмов распознавания или регрессионных моделей. Следует отметить тесную связь между обоими подходами.

Пример

Пусть $X_1, X_2, \ldots, X_m$ – независимые одинаково распределённые случайные величины (н.о.р.с.в), причём

$$X_i \sim \mathcal{N}(x_i|\theta, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x_i-\theta)^2}{2\sigma^2}\right).$$

Требуется с помощью метода максимального правдоподобия оценить значение θ.

Пример

Пусть $X_1, X_2, \ldots, X_m$ – независимые одинаково распределённые случайные величины (н.о.р.с.в), причём

$$X_i \sim \mathcal{N}(x_i|\theta, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x_i-\theta)^2}{2\sigma^2}\right).$$

Требуется с помощью метода максимального правдоподобия оценить значение θ.

Запишем совместное распределение величин $X_1, \ldots, X_m$:

$$p(\boldsymbol{x}|\theta, \sigma^2) = p(x_1, x_2, \ldots, x_m|\theta, \sigma^2) = \{\text{Нез-ть}\} =$$
$$= \prod_{j=1}^{m} p(x_i|\theta, \sigma^2) = \prod_{j=1}^{m} \mathcal{N}(x_j|\theta, \sigma^2).$$

Продолжение примера I

Функция $p(\boldsymbol{x}|\theta, \sigma^2)$:

- как функция от $\boldsymbol{x}$ – условная плотность, в частности, $\int p(\boldsymbol{x}|\theta, \sigma^2)d\boldsymbol{x} = 1$;
- как функция от θ, σ^2 – функция правдоподобия.

Оценка θ с помощью максимизации правдоподобия соответствует задаче оптимизации:

$$L(\theta, \sigma^2) = p(\boldsymbol{x}|\theta, \sigma^2) \to \max_{\theta}.$$

При работе с функционалами в форме произведений удобно переходить к логарифму:

$$\theta_{ML} = \arg\max_{\theta} L(\theta, \sigma^2) = \arg\max_{\theta} \log L(\theta, \sigma^2) =$$

$$= \arg\max_{\theta} \left[-\frac{1}{2\sigma^2} \sum_{j=1}^{m} (x_j - \theta)^2 \underbrace{-\frac{m}{2} \log 2\pi - m \log \sigma}_{const} \right]$$

Продолжение примера II

Дифференцируя $\log L(\theta, \sigma^2)$ по θ и приравнивая производную к нулю, получаем:

$$\frac{\partial}{\partial \theta} \log L(\theta, \sigma^2) = -\frac{1}{2\sigma^2} \left(2m\theta - 2 \sum_{j=1}^{m} x_j \right) = 0.$$

Отсюда

$$\theta_{ML} = \frac{1}{m} \sum_{j=1}^{m} x_j.$$

Заметим, что оценка θ_{ML} не зависит от дисперсии σ^2.

Поиск оптимальных алгоритмов прогнозирования и распознавания

Для подавляющего числа приложений ни общий вид распределений, ни значения конкретных их параметров неизвестны. В связи с этим возникло большое число разнообразных подходов к решению задач прогнозирования, использование которых позволяло добиваться определённых успехов при решении конкретных задач.

Методы прогнозирования

- Статистические методы
- Линейные модели регрессионного анализа
- Различные методы, основанные на линейной разделимости
- Методы, основанные на ядерных оценках
- Нейросетевые методы
- Комбинаторно-логические методы и алгоритмы вычисления оценок
- Алгебраические методы
- Решающие или регрессионные деревья и леса
- Методы, основанные на опорных векторах

Эмпирические методы оценки обобщающей способности

Обобщающая способность может оцениваться по случайной выборке объектов из одной и той же генеральной совокупности, соответствующей исследуемому процессу, которую принято называть контрольной выборкой. Контрольная выборка не должна содержать объекты из обучающей выборки.

Контрольная выборка имеет вид $\tilde{S}_c = \{(y_1, \boldsymbol{x}_1), \ldots, (y_{m_c}, \boldsymbol{x}_{m_c})\}$, где

y_j – значение переменной Y для j-го объекта;

$\boldsymbol{x}_j$ – значение вектора переменных $X_1, \ldots, X_n$ для j-го объекта;

m_c – число объектов в $\tilde{S}_c$.

Эмпирические методы оценки обобщающей способности

Обобщающая способность A может оцениваться с помощью функционала риска

$$Q(\tilde{S}_c, A) = \frac{1}{m_c} \sum_{i=1}^{m_c} \lambda[y_j, A(\boldsymbol{x}_j)].$$

При $m_c \to \infty$ согласно закону больших чисел

$$Q(\tilde{S}_c, A) \to E_\Omega\{\lambda[Y, A(\boldsymbol{x})]\}.$$

Эмпирические методы оценки обобщающей способности

Обычно при решении задачи прогнозирования по прецедентам в распоряжении исследователей сразу оказывается весь массив существующих эмпирических данных $\tilde{S}_{in}$. Для оценки точности прогнозирования могут быть использованы следующие стратегии:

1. Выборка $\tilde{S}_{in}$ случайным образом расщепляется на выборку $\tilde{S}_t$ для обучения алгоритма прогнозирования и выборку $\tilde{S}_c$ для оценки точности;
2. Процедура кросс-проверки. Выборка $\tilde{S}_{in}$ случайным образом расщепляется на выборки $\tilde{S}_A$ и $\tilde{S}_B$. На первом шаге $\tilde{S}_A$ используется для обучения и $\tilde{S}_B$ для контроля. На следующем шаге $\tilde{S}_A$ и $\tilde{S}_B$ меняются местами.

3. Процедура скользящего контроля выполняется по полной выборке $\tilde{S}_{in}$ за $m = |\tilde{S}_{in}|$ шагов. На j-ом шаге формируется обучающая выборка $\tilde{S}_t^j = \tilde{S}_{in} \setminus s_j$, где $s_j = (y_j, \boldsymbol{x}_j)$ – j-ый объект $\tilde{S}_{in}$, и контрольная выборка $\tilde{S}_c$, состоящая из единственного объекта s_j. Процедура скользящего контроля вычисляет оценку обобщающей способности как

$$Q_{sc}(\tilde{S}_{in}, A) = \frac{1}{m} \sum_{j=1}^{m} \lambda[y_j, A(\boldsymbol{x}_j, \tilde{S}_t^j)].$$

Несмещённость оценки скользящего контроля

Под несмещённостью оценки скользящего контроля понимается выполнение следующего равенства

$$E_{\Omega_m}\{Q_{sc}(\tilde{S}_m, A)\} = E_{\Omega_{m-1}} E_{\Omega}\{\lambda[Y, A(\boldsymbol{x}, \tilde{S}_{m-1})]\}.$$

Покажем, что несмещённость имеет место, если выборка $\tilde{S}_{in}$ является независимой выборкой объектов из генеральной совокупности Ω.

Несмещённость оценки скользящего контроля

Напомним, что в этом случае $\tilde{S}_{in}$ является элементом вероятностного пространства $\langle \Omega_m, \Sigma_m, P_m \rangle$. Произвольная подвыборка $\tilde{S}_{in}$ размером $m' < m$ с произвольным порядком объектов является элементом вероятностного пространства $\langle \Omega_{m'}, \Sigma_{m'}, P_{m'} \rangle$, которое строится также, как и вероятностное пространство $\langle \Omega_m, \Sigma_m, P_m \rangle$.

Несмещённость оценки скользящего контроля

$$E_{\Omega_m}\{Q_{sc}(\tilde{S}_m, A)\} = E_{\Omega_m}\left\{\frac{1}{m}\sum_{j=1}^{m}\lambda[y_j, A(\boldsymbol{x}_j, S_t^j)]\right\} =$$

$$\frac{1}{m}\sum_{j=1}^{m}E_{\Omega_m}\lambda[y_j, A(\boldsymbol{x}_j, S_t^j)].$$

Однако из ранее сказанного следует, что $\forall j$ выборка $\tilde{S}_t^j$ является элементом пространства Ω_{m-1}. Объект $(y_j, \boldsymbol{x}_j)$ является элементом Ω.

Несмещённость оценки скользящего контроля

Из упомянутых свойств, а также из теоремы Фубини следует

$$E_{\Omega_m}\{\lambda[y_j, A(\boldsymbol{x}_j, \tilde{S}_t^j)]\} = E_{\Omega_{m-1}} E_{\Omega}\{\lambda[Y, A(\boldsymbol{x}, S_{m-1})]\}.$$

Таким образом,

$$E_{\Omega_m}\{Q_{sc}[\tilde{S}_m, A]\} = \frac{1}{m}\sum_{i=1}^{m} E_{\Omega_{m-1}} E_{\Omega}\{\lambda[Y, A(\boldsymbol{x}, \tilde{S}_{m-1})]\} =$$

$$E_{\Omega_{m-1}} E_{\Omega}\{\lambda[Y, A(\boldsymbol{x}, \tilde{S}_{m-1})]\}.$$

Содержание лекции

Линейная регрессия

Распространённым средством решения задач прогнозирования непрерывной величины Y по переменным $X_1, \ldots, X_n$ является использование метода множественной линейной регрессии. В данном методе связь переменной Y с переменными $X_1, \ldots, X_n$ задаётся с помощью линейной модели

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n + \varepsilon, \tag{1}$$

где $\beta_0, \beta_1, \ldots, \beta_n$ - вещественные регрессионные коэффициенты, ε - случайная величина, являющаяся ошибкой прогнозирования. Регрессионные коэффициенты ищутся по обучающей выборке

$$\tilde{S}_t = \{s_1 = (y_1, \mathbf{x}_1), \ldots, s_m = (y_m, \mathbf{x}_m)\}, \tag{2}$$

где $\mathbf{x}_j = (x_{j1}, \ldots, x_{jn})$ вектор значений переменных $X_1, \ldots, X_n$ для объекта s_j.

Линейная регрессия. Метод наименьших квадратов

Традиционным способом поиска регрессионных коэффициентов является метод наименьших квадратов (МНК). МНК заключается в минимизации функционала эмпирического риска с квадратичными потерями

$$Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n) = \sum_{j=1}^{m} [y_j - \beta_0 - \sum_{i=1}^{n} x_i \beta_{ij}]^2 \tag{3}$$

То есть оценки $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_n$ регрессионных коэффициентов $\beta_0, \beta_1, \ldots, \beta_n$ по методу МНК удовлетворяют условию минимума функционала эмпирического риска с квадратичными потерями

$$(\hat{\beta}_0, \ldots, \hat{\beta}_n) = \arg\min[Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)]. \tag{4}$$

Линейная регрессия. Метод наименьших квадратов

Предположим взаимосвязь между величиной Y и переменными $X_1, \ldots, X_n$ описывается выражением

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n + \varepsilon_N(0, \sigma) \tag{5}$$

Где ошибка $\varepsilon_N(0, \sigma)$ распределена нормально, При ‘этом дисперсия ошибки σ^2 не зависит от $X_1, \ldots, X_n$, а математическое ожидание ошибки равно 0 при произвольных значениях прогностических переменных: $E_\Omega(\varepsilon_N \mid \boldsymbol{x}) = 0$, $E_\Omega(\varepsilon_N^2 \mid \boldsymbol{x}) = \sigma^2$ при произвольном допустимом векторе $\boldsymbol{x}$. В этом случае метод МНК тождественен более общему статистическому методу оценивания параметров статистических распределений – Методу максимального правдоподобия (ММП).

Линейная регрессия. Метод максимального правдоподобия

Метод максимального правдоподобия. Предположим, что некоторое пространство событий с заданной на нём вероятностной мерой P характеризуется переменными $Z_1, \ldots, Z_d$. Метод ММП позволяет восстанавливать плотность распределения вероятностей по случайным выборкам, если общий вид плотности вероятностного распределения известен. Пусть плотность распределения принадлежит семейству функций, задаваемому вектором параметров $\theta_1, \ldots, \theta_r$, принимающем значения из множества $\widetilde{\Theta}$:

$$\{p(Z_1, \ldots, Z_d, \theta_1, \ldots, \theta_r) \mid \boldsymbol{\theta} \in \widetilde{\Theta}\}.$$

Предположим, что у нас имеется случайная выборка объектов, описываемых векторами $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m$ переменных $Z_1, \ldots, Z_d$.

Линейная регрессия. Метод максимального правдоподобия.

Напомним, что метод МП заключается в выборе в семействе

$$\{p(Z_1, \ldots, Z_d, \theta_1, \ldots, \theta_r) \mid \boldsymbol{\theta} \in \widetilde{\Theta}\}$$

плотности,для которой достигает максимума функция правдоподобия

$$L(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m, \theta_1, \ldots, \theta_r) = \prod_{j=1}^{m} p(\boldsymbol{z}_j, \boldsymbol{\theta}). \tag{6}$$

Иными словами оценка $\hat{\boldsymbol{\theta}}$ вектора параметров $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_r)$ вычисляется как

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta} \in \widetilde{\Theta}} [L(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m, \theta_1, \ldots, \theta_r)]. \tag{7}$$

Попытаемся вычислить значения параметров $(\beta_0, \beta_1, \ldots, \beta_n)$ исходя из предположения (5). Согласно (5) разность $Y - \beta_0 - \beta_1 X_1 - \ldots - \beta_n X_n$ годчиняется нормальному распределению с нулевым математическим ожиданием и дисперсией σ^2.

Линейная регрессия. Связь между МНК и ММП.

Плотность распределения в пространстве переменных $Y, X_1, \ldots, X_n$ может быть восстановлена по обучающей выборке $\tilde{S}_t = \{(y_1, \mathbf{x}_1), \ldots, (y_m, \mathbf{x}_m)\}$, путём максимизации функции правдоподобия

$$L(\tilde{S}_t, \beta_0, \ldots, \beta_n) = \prod_{j=1}^{m} \frac{1}{(\sqrt{2\pi}\sigma)} \exp \frac{-(y_j - \beta_0 - \sum_{i=1}^{n} \beta_i x_{ji})^2}{2\sigma^2} \tag{8}$$

Очевидно, точка экстремума функции правдоподобия $L(\tilde{S}_t, \beta_0, \ldots, \beta_n)$ совпадает с точкой экстремума функции

$$\ln[L(\tilde{S}_t, \beta_0, \ldots, \beta_n)] = -\sum_{j=1}^{m} [\frac{1}{2} \ln 2\pi + \ln \sigma + \frac{1}{2\sigma^2} (y_j - \beta_0 - \sum_{i=1}^{n} \beta_i x_{ji})^2] \tag{9}$$

Линейная регрессия. Связь между МНК и ММП.

Однако точка максимума $\ln[L(\tilde{S}_t, \beta_0, \ldots, \beta_n)]$ совпадает с точкой минимума функции $Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)$, оптимизируемой в методе МНК, что позволяет сделать вывод о эквивалентности ММП и МНК

Линейная регрессия. Одномерная модель.

Рассмотрим простейший вариант линейной регрессии, описывающей связь между переменной Y и единственной переменной X : $Y = \beta_0 + \beta_1 X + \varepsilon$. Функционал эмпирического риска на выборке $\tilde{S}_t = \{(y_1, x_1), \ldots, (y_m, x_m)\}$ принимает вид

$$Q(\tilde{S}_t, \beta_0, \beta_1) = \frac{1}{m} \sum_{j=1}^{m} [y_j - \beta_0 - x_j \beta_1]^2. \tag{10}$$

Необходимым условием минимума функционала $Q(\tilde{S}_t, \beta_0, \beta_1)$ является выполнение системы из двух уравнений

$$\frac{\partial Q(\tilde{S}_t, \beta_0, \beta_1)}{\partial \beta_0} = -\frac{2}{m} \sum_{j=1}^{m} y_j + 2\beta_0 + \frac{2\beta_1}{m} \sum_{j=1}^{m} x_j = 0 \tag{11}$$

$$\frac{\partial Q(\tilde{S}_t, \beta_0, \beta_1)}{\partial \beta_1} = -\frac{2}{m} \sum_{j=1}^{m} x_j y_j + \frac{2\beta_0}{m} \sum_{j=1}^{m} x_j + \frac{2\beta_1}{m} \sum_{j=1}^{m} x_j^2 = 0$$

Оценки $\hat{\beta}_0, \hat{\beta}_1$ являются решением системы (11) относительно параметров соответственно β_0, β_1 . Оценки регрессионных коэффициентов могут быть записаны в виде

$$\hat{\beta}_1 = \frac{\sum_{j=1}^{m} x_j y_j - \frac{1}{m}\sum_{j=1}^{m} y_j \sum_{j=1}^{m} x_j}{\sum_{j=1}^{m} x_j^2 - \frac{1}{m}(\sum_{j=1}^{m} x_j)^2}, \tag{12}$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}_1 \overline{x}$$

, где $\overline{y} = \frac{1}{m}\sum_{j=1}^{m} y_j$, $\overline{x} = \frac{1}{m}\sum_{j=1}^{m} x_j$. Выражение для $\hat{\beta}_1$ может быть переписано в виде

$$\hat{\beta}_1 = \frac{Cov(Y, X \mid \tilde{S}_t)}{D(X \mid \tilde{S}_t}, \tag{13}$$

где $Cov(Y, X \mid \tilde{S}_t)$ является выборочной ковариацией переменных Y и X, $D(X \mid \tilde{S}_t)$ является выборочной дисперсией переменной X.

То есть

$$Cov(Y, X \mid \tilde{S}_t) = \frac{1}{m} \sum_{j=1}^{m} (y_j - \overline{y})(x_j - \overline{x})$$

$$D(X \mid \tilde{S}_t) = \frac{1}{m} \sum_{j=1}^{m} (x_j - \overline{x})^2$$

Многомерная регрессия

При вычислении оценки вектора параметров $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_n)$ в случае многомерной линейной регрессии удобно использовать матрицу плана $\mathbf{X}$ размера $m \times (n+1)$, которая строится по обучающей выборке $\tilde{S}_t$. Матрица плана имеет вид

$$\mathbf{X} = \begin{pmatrix} 1 & \mathrm{x}_{11} & \ldots & \mathrm{x}_{1n} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \mathrm{x}_{j1} & \ldots & \mathrm{x}_{jn} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \mathrm{x}_{m1} & \ldots & \mathrm{x}_{mn} \end{pmatrix}.$$

То есть j-я строка матрицы плана представляет собой вектор значений переменных $X_1, \ldots, X_n$ для объекта s_j с одной добавленной слева компонентой, содержащей 1.

Многомерная регрессия

Пусть $\mathbf{y} = (y_1, \ldots, y_m)$ - вектор значений переменной Y . Связь Y с переменными $X_1, \ldots, X_n$ на объектах обучающей выборки может быть описана с помощью матричного уравнения

$$\mathbf{y} = \boldsymbol{\beta}\mathbf{X}^t + \boldsymbol{\varepsilon},$$

где $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_m)$ - вектор ошибок прогнозирования для объектов $\tilde{S}_t$. Функционал $Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)$ может быть записан в виде

$$Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n) = \sum_{j=1}^{m} [y_j - \beta_0 - \sum_{i=1}^{n} \beta_i \breve{x}_{ji}]^2, \qquad (14)$$

где $\breve{x}_{ji}$ - элементы матрицы плана $\mathbf{X}$, определяемые равенствами $\breve{x}_{j1} = 1$, $\breve{x}_{ji} = x_{j(i-1)}$ при $i > 1$.

Необходимым условием минимума функционала $Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)$ является выполнение системы из $n+1$ уравнений

$$\frac{\partial Q(\tilde{S}_t, \beta_0, \ldots, \beta_n)}{\partial \beta_0} = -\frac{2}{m}[\sum_{j=1}^{m} y_j \breve{x}_{j1} - \sum_{j=1}^{m}\sum_{i=1}^{n+1} \beta_i \breve{x}_{ji} \breve{x}_{j1}] = 0 \qquad (15)$$

$$\ldots, \ldots, \ldots, \ldots, \ldots, \ldots, \ldots, \ldots$$

$$\frac{\partial Q(\tilde{S}_t, \beta_0, \ldots, \beta_n)}{\partial \beta_n} = -\frac{2}{m}[\sum_{j=1}^{m} y_j \breve{x}_{jn} - \sum_{j=1}^{m}\sum_{i=1}^{n+1} \beta_i \breve{x}_{ji} \breve{x}_{jm}] = 0$$

Вектор оценок значений регрессионных коэффициентов $\hat{\beta}_0, \ldots, \hat{\beta}_n$ является решением системы уравнений (15) .

В матричной форме система (15) может быть записана в виде

$$-2\mathbf{X}^t\boldsymbol{y}^t + 2\mathbf{X}^t\mathbf{X}\boldsymbol{\beta}^t = 0 \tag{16}$$

Решение системы (16) существует, если $\det(\mathbf{X}^t\mathbf{X}) \neq 0$. В этом случае для $\mathbf{X}^t\mathbf{X}$ существует обратная матрица и решение (16) относительно вектора $\boldsymbol{\beta}$ может быть записано в виде:

$$\hat{\boldsymbol{\beta}}^t = (\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\boldsymbol{y}^t. \tag{17}$$

Из теории матриц следует, что $\det(\mathbf{X}^t\mathbf{X}) = 0$ если ранг матрицы $\mathbf{X}$ по строкам менее $(n+1)$, что происходит, если m-мерный вектор значений одной из переменных $X_{i'} \in \{X_1, \ldots, X_n\}$ на выборке $\tilde{S}_t$ является линейной комбинаций m-мерных векторов значений на $\tilde{S}_t$ других переменных из $\{X_1, \ldots, X_n\}$.

Многомерная регрессия

При сильной коррелированности одной из переменных $X_{i'} \in \{X_1, \ldots, X_n\}$ на выборке с какой-либо линейной комбинацией других переменных значение $\det(\mathbf{X}^t\mathbf{X})$ оказывается близким к 0. При этом вычисленный вектор оценок $\hat{\boldsymbol{\beta}}^t$ может сильно изменяться при относительно небольших чисто случайных изменениях вектора $\boldsymbol{y} = (y_1, \ldots, y_m)$. Данное явление называется мультиколлинеарностью. Оценивание регрессионных коэффициентов с использованием МНК при наличии мультиколлинеарности оказывается неустойчивым. Отметим также, что $\det(\mathbf{X}^t\mathbf{X}) = 0$ при $n + 1 > m$.Поэтому МНК не может использоваться для оценивания регрессионных коэффициентов, когда число переменных превышает число объектов в обучающей выборке. На практике высокая устойчивость достигается только, когда число объектов в выборках по крайней мере в 3-5 раз превышает число переменных.

Свойства оптимальных регрессий

Рассмотрим свойства линейных регрессий, минимизирующих квадрат ошибки на пространстве событий Ω . Пусть $R(X_1, \ldots, X_n)$-регрессионная функция, которая не может быть улучшена с помощью дополнительного линейного преобразования. Иными словами при произвольных α_0 и α_1

$$E_\Omega[Y - \alpha_0 - \alpha_1 R]^2 \geq E_\Omega[Y - R]^2 \tag{18}$$

То есть минимум $E_\Omega[Y - \alpha_0 - \alpha_1 R]^2$ достигается при $_0 = 0$ и $\alpha_1 = 1$. Необходимым условием минимума $E_\Omega[Y - \alpha_0 - \alpha_1 R]^2$ является равенство 0 частных производных

$$\frac{\partial E_\Omega[Y - \alpha_0 - \alpha_1 R]^2}{\partial \alpha_0} = 0, \tag{19}$$

$$\frac{\partial E_\Omega[Y - \alpha_0 - \alpha_1 R]^2}{\partial \alpha_1} = 0.$$

Свойства оптимальных регрессий

Проведём простейшие преобразования

$$E_\Omega[Y - \alpha_0 - \alpha_1 R]^2 = E_\Omega Y^2 - 2\alpha_0 E_\Omega Y - 2\alpha_1 E_\Omega Y R + +$$

$$+\alpha_1^2 E_\Omega R^2 + 2\alpha_0\alpha_1 E_\Omega R + \alpha_0^2.$$

Откуда следует, что уравнения (19) эквивалентны уравнениям

$$2\alpha_1 E_\Omega R + 2\alpha_0 - 2\alpha_1 E_\Omega Y = 0, \tag{20}$$

$$-2E_\Omega(YR) + 2\alpha_1 E_\Omega R^2 + 2\alpha_0 E_\Omega R = 0.$$

Принимая во внимание, что в точке экстремума $\alpha_0 = 0$ и $\alpha_1 = 1$ получаем из системы (20) следующие свойства оптимального линейного прогнозирующего алгоритма 1) $E_\Omega R = E_\Omega Y$, 2)$E_\Omega(YR) = E_\Omega R^2$,

Свойства оптимальных регрессий

Из свойств 1) 2) следует , что дисперсия R равна ковариации Y и R. Действительно,

$$D(R) = E_\Omega(R - E_\Omega R)^2 = E_\Omega R^2 - (E_\Omega R)^2$$

. Однако

$$cov(Y, R) = E_\Omega(R - E_\Omega R)(Y - E_\Omega Y) = E_\Omega Y R - E_\Omega R^2$$

. То есть

$$3) cov(Y, R) = D(R).$$

Рассмотрим теперь коэффициент корреляции между Y и R - $\rho(Y, R)$.

$$\rho(Y, R) = \frac{cov(Y, R)}{\sqrt{D(Y)D(R)}} = \sqrt{\frac{D(R)}{D(Y)}}.$$

Свойства оптимальных регрессий

Рассмотрим величину ошибки прогнозирования Y с помощью R.

$$4)\Delta(Y,R) = E_\Omega(Y-R)^2 = E_\Omega Y^2 - 2E_\Omega(YR) + E_\Omega R^2 =$$

$$= E_\Omega Y^2 - E_\Omega R^2 = E_\Omega Y^2 - (E_\Omega Y)^2 + (E_\Omega Y)^2 - E_\Omega R^2 =$$

$$= E_\Omega Y^2 - E_\Omega R^2 = E_\Omega Y^2 - (E_\Omega Y)^2 + (E_\Omega R)^2 - E_\Omega R^2 = D(Y) - D(R).$$

Из свойств (3) и (4) легко следует свойство для относительной ошибки $\Delta_{rel}(Y,R) = \frac{\Delta(Y,R)}{D(Y)}$:

$$5)\Delta_{rel}(Y,R) = 1 - \rho^2(Y,R).$$

Структура ошибки прогнозирования. Обобщённая ошибка.

Напомним, что обобщающая способность алгоритма прогнозирования A, обученного по выборке $\widetilde{S}_t$ с помощью некоторого метода **M** измеряется величиной потерь на генеральной совокупности Ω:

$$E_\Omega \lambda[Y, A(\boldsymbol{x}, \widetilde{S}_t)].$$

Для оценки эффективности использования метода прогнозирования **M** для прогнозирования случайного процесса, связанного с генеральной совокупностью Ω, при фиксированном размере обучающей выборки m естественно использовать математическое ожидание потерь по пространству всевозможных обучающих выборок длины m:

$$\Delta_G = E_{\Omega_m} E_\Omega \lambda[Y, A(\boldsymbol{x}, \widetilde{S}_t)],$$

где Ω_m - рассмотренное ранее пространство обучающих выборок длины m.

Структура ошибки прогнозирования. Обобщённая ошибка.

Величину Δ_G будем называть обобщёнными потерями. При использовании в качестве функции потерь квадрата ошибки обобщённые потери становятся обобщённой квадратичной ошибкой и принимают вид

$$\Delta_G = E_{\Omega_m} E_{\Omega}[Y - A(\boldsymbol{x}, \widetilde{S}_t)]^2.$$

Проведём простые преобразования:

$$E_{\Omega_m} E_{\Omega}[Y - A(\boldsymbol{x}, \widetilde{S}_t)]^2 = E_{\Omega_m} E_{\Omega}[Y - E(Y \mid \boldsymbol{x}) +$$

$$+ E(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2 = E_{\Omega_m} E_{\Omega}[Y - E(Y \mid \boldsymbol{x})]^2 +$$

$$+ E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2 +$$

$$+ 2E_{\Omega_m} E_{\Omega}\{[E(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)][Y - E(Y \mid \boldsymbol{x})]\}.$$

Структура обобщённой ошибки.

Покажем справедливость равенства

$$E_{\Omega_m} E_\Omega\{[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)][Y - E(Y \mid \boldsymbol{x})]\} = 0 \qquad (21)$$

Действительно

$$E_\Omega\{[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)][Y - E(Y \mid \boldsymbol{x})]\} =$$

$$= \int_{\widetilde{X}} E_\Omega\{[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)][Y - E(Y \mid \boldsymbol{x})] \mid \boldsymbol{x}\} p(\boldsymbol{x}) dX_1 \ldots dX_n,$$

где $\widetilde{X}$ совместная область допустимых значений $X_1, \ldots, X_n$ в пространстве $\mathbb{R}^n$, $p(\boldsymbol{x})$ - плотность вероятности внутри $\widetilde{X}$. При любом фиксированном $\boldsymbol{x}$ множитель $[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]$ не зависит от объекта, для которого производится прогнозирование, и может быть вынесен за знак математического ожидания:

Структура обобщённой ошибки.

$$E_\Omega\{[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)][Y - E(Y \mid \boldsymbol{x})] \mid \boldsymbol{x}\} =$$
$$= [E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]E_\Omega\{[Y - E_\Omega(Y \mid \boldsymbol{x})] \mid \boldsymbol{x}\}.$$

Однако по определению условного математического ожидания при любом $\boldsymbol{x}$

$$E_\Omega\{[Y - E_\Omega(Y \mid \boldsymbol{x})] \mid \boldsymbol{x}\} = 0.$$

Откуда следует справедливость равенства (21). Принимая во внимание, что $[Y - E_\Omega(Y \mid \boldsymbol{x})]$ не зависит от $\widetilde{S}_t$ получаем

$$E_{\Omega_m} E_\Omega[Y - E(Y \mid \boldsymbol{x})]^2 = E_\Omega[Y - E(Y \mid \boldsymbol{x})]^2$$

. В итоге

$$\Delta_G = E_\Omega[Y - E(Y \mid \boldsymbol{x})]^2 + E_{\Omega_m} E_\Omega[E_\Omega(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2.$$

Введём обозначение

$$\hat{A}(\boldsymbol{x}) = E_{\Omega_m} A(\boldsymbol{x}, \widetilde{S}_t)$$

Структура обобщённой ошибки.

Компонента разложения

$$E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2$$

может быть представлена в виде

$$E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x}) + \hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2 =$$

$$= E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})]^2 + E_{\Omega_m} E_{\Omega}[\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2 +$$

$$+2E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})][\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]$$

Справедливо равенство

$$2E_{\Omega_m} E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})][\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)] = 0. \qquad (22)$$

Структура обобщённой ошибки.

Действительно

$$2E_{\Omega_m}E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})][\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)] =$$

$$2E_{\Omega}\{[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})]E_{\Omega_m}[\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]\}.$$

Однако из определения $\hat{A}(\boldsymbol{x})$ следует, что

$$E_{\Omega}[\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)] = 0.$$

Поэтому равенство (22) справедливо. В итоге справедливо трёхкомпонентное разложение обобщённой квадратичной ошибки Δ_G:

$$\Delta_G = E_{\Omega}[Y - E(Y \mid \boldsymbol{x})]^2 + E_{\Omega}[E_{\Omega}(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})]^2 + \qquad (23)$$

$$+E_{\Omega_m}E_{\Omega}[\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2 =$$

$$= \Delta_N + \Delta_B + \Delta_V$$

Структура обобщённой ошибки.

Шумовая компонента.

$$\Delta_N = E_\Omega[Y - E(Y \mid \boldsymbol{x})]^2$$

является минимально достижимой квадратичной ошибкой прогноза, которая не может быть устранена с использованием только математических средств. Составляющая сдвига (Bias).

$$\Delta_B = E_\Omega[E_\Omega(Y \mid \boldsymbol{x}) - \hat{A}(\boldsymbol{x})]^2$$

Высокое значение компоненты сдвига соответствует отсутствию в модели $\widetilde{M} = \{A : \widetilde{X} \to \widetilde{Y}$, внутри которой осуществляется поиск, алгоритмов, достаточно хорошо аппроксимирующих объективно существующую зависимость Y от переменных $X_1, \ldots, X_n$. Составляющая сдвига может быть снижена, например, путём расширения модели за счёт включения в него дополнительных более сложных алгоритмов, что обычно позволяет повысить точность аппроксимации данных.

Структура обобщённой ошибки.

Дисперсионная составляющая (Variance).

$$\Delta_V = E_{\Omega_m} E_{\Omega}[\hat{A}(\boldsymbol{x}) - A(\boldsymbol{x}, \widetilde{S}_t)]^2$$

характеризует неустойчивость обученных прогнозирующих алгоритмов при статистически возможных изменениях в обучающих выборках. Дисперсионная составляющая возрастает при небольших размерах обучающей выборки. Дисперсионная составляющая может быть снижена путём выбора сложности модели, соответствующей размеру обучающих данных.

Структура обобщённой ошибки.

Таким образом существует Bias-Variance дилемма Составляющая сдвига может быть снижена путём увеличения разнообразия модели. Однако увеличение разнообразия модели при недостаточном объёме обучающих данных ведёт к росту компоненты сдвига. Наиболее высокая точность прогноза достигается, при поддержании правильного баланса между разнообразием используемой модели и объёмом обучающих данных

Методы регуляризации

Одним из возможных способов борьбы с неустойчивостью является использование методов, основанных на включение в исходный оптимизируемый функционал $Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)$ дополнительной штрафной компоненты. Введение такой компоненты позволяет получить решение, на котором $Q(\tilde{S}_t, \beta_0, \beta_1, \ldots, \beta_n)$ достаточно близок к своему глобальному минимуму. Однако данное решение оказывается значительно более устойчивым и благодаря устойчивости позволяет достигать существенно более высокой обобщающей способности. Подход к получению более эффективных решений с помощью включения штрафного слагаемого в оптимизируемый функционал принято называть регуляризацией по Тихонову.

На первом этапе переходим от исходных переменных $\{X_1, \ldots, X_n\}$ к стандартизированным $\{X_1^s, \ldots, X_n^s\}$, где $X_i^s = \frac{X_i - \hat{X}_i}{\hat{\sigma}_i}$, $\hat{X}_i = \frac{1}{m}\sum_{j=1}^{m} x_{ji}$, $\hat{\sigma}_i = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(\hat{X}_i - x_{ji}^2)}$, а также от исходной прогнозируемой переменной Y к стандартизованной прогнозируемой переменной $Y^s = Y - \frac{1}{m}\sum_{j=1}^{m} y_j$. Пусть $\breve{x}_{j1}^s = 1$, $\breve{x}_{ji}^s = x_{j(i-1)}^s$ при $i > 1$, где $x_{j(i-1)}^s$ - значение признака X_i^s для j-го объекта. Пусть также $\mathbf{X}_s = \begin{pmatrix} 1 & \mathrm{x}_{11}^s & \ldots & \mathrm{x}_{1n}^s \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \mathrm{x}_{j1}^s & \ldots & \mathrm{x}_{jn}^s \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \mathrm{x}_{m1}^s & \ldots & \mathrm{x}_{mn}^s \end{pmatrix}$. - матрица плана для стандартизированных переменных, $\boldsymbol{y}^s = (y_1^s, \ldots, y_m^s)$ - вектор значений стандартизованной переменной Y_s .

Одним из первых методов регрессии, использующих принцип регуляризации, является метод гребневой регрессии (ridge regression). В гребневой регрессии в оптимизируемый функционал дополнительно включается сумма квадратов регрессионных коэффициентов при переменных $\{X_1^s, \ldots, X_n^s\}$. В результате функционал имеет вид

$$Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n) = \frac{1}{m}\sum_{j=1}^{m}[y_j - \beta_0 - \sum_{i=1}^{n}\beta_i \breve{x}_{ji}^s]^2 + \gamma\sum_{i=1}^{n}\beta_i^2, \quad (24)$$

где γ - положительный вещественный параметр. Пусть $\boldsymbol{\beta}_r$ является вектором оценок регрессионных коэффициентов, полученным в результате минимизации $Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n)$.

Методы регуляризации

Отметим, что увеличение регрессионных коэффициентов приводит к увеличению $Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n)$. Таким образом использование гребневой регрессии приводит к снижению длины вектора регрессионных коэффициентов при переменных $\{X_1^s, \ldots, X_n^s\}$. Рассмотрим конкретный вид вектора регрессионных коэффициентов $\boldsymbol{\beta}_r$ в гребневой регрессии. Необходимым условием минимума функционала $Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n)$ является выполнение системы из $n+1$ уравнений:

$$\frac{\partial Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n)}{\partial \beta_0} = -\frac{2}{m}[\sum_{j=1}^{m} y_j \breve{x}_{j1}^s - \sum_{j=1}^{m}\sum_{i=1}^{n+1} \beta_i \breve{x}_{ji}^s \breve{x}_{j1}^s] + 2\gamma\beta_0 = 0 \quad (25)$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$\frac{\partial Q_{ridge}(\tilde{S}_t, \beta_0, \ldots, \beta_n)}{\partial \beta_n} = -\frac{2}{m}[\sum_{j=1}^{m} y_j \breve{x}_{jn}^s - \sum_{j=1}^{m}\sum_{i=1}^{n+1} \beta_i \breve{x}_{ji}^s \breve{x}_{jm}^s] + 2\gamma\beta_n = 0$$

Поэтому вектор оценок регрессионных коэффициентов в методе гребневая регрессия является решением системы (25).

В матричной форме система (25) может быть записана в виде

$$-2\mathbf{X}_s^t\boldsymbol{y}_s^t + (2\mathbf{X}_s^t\mathbf{X}_s + 2\gamma\mathbf{I})\boldsymbol{\beta}_r^t = 0 \tag{26}$$

или в виде

$$\boldsymbol{\beta}_r^t = \mathbf{X}_s^t\boldsymbol{y}_s^t(\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I})^{-1} \tag{27}$$

где $\mathbf{I}$ - единичная матрица. Отметим, что произведение $\mathbf{X}_s^t\mathbf{X}_s$ представляет собой симметрическую неотрицательно определённую матрицу. Матрица $\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I}$ также является симметрической матрицей. Каждому собственному значению λ_k матрицы $\mathbf{X}_s^t\mathbf{X}_s$ соответствует собственное значение $\lambda_k + \gamma$ матрицы $\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I}$. Пусть λ_{min}^{γ} минимальное собственное значение матрицы $\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I}$ удовлетворяет неравенству $\lambda_{min}^{\gamma} > \gamma$. Откуда следует, что всегда $\det(\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I}) > 0$, а обратная матрица $(\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I})^{-1}$ всегда существует. Достаточно большая величина $\det(\mathbf{X}_s^t\mathbf{X}_s + \gamma\mathbf{I})$ приводит к относительно небольшим изменениям оценок регрессионных коэффициентов при небольших изменениях в обучающих выборках.

Наряду с гребневой регрессией в последние годы получил распространение метод Лассо, основанный на минимизации функционала

$$Q_{lasso}(\tilde{S}_t, \beta_0, \ldots, \beta_n) = \sum_{j=1}^{m}[y_j - \beta_0 - \sum_{i=1}^{n}\beta_i \breve{x}_{ji}^s]^2 + \gamma \sum_{i=1}^{n} \mid \beta_i \mid. \quad (28)$$

Интересной особенностью метода Лассо является равенство 0 части из регрессионных коэффициентов . Однако равенство 0 коэффициента на самом деле означает исключение из модели соответствующей ему переменной. Поэтому метод Лассо не только строит оптимальную регрессионную модель, но и производит отбор переменных. Метод может быть использован для отбора переменных в условиях, когда размерность данных превышает размер выборки. Отметим, что общее число отобранных переменных не может превышать размера обучающей выборки . Эксперименты показали, что эффективность отбора переменных методом Лассо снижается, при высокой взаимной корреляции некоторых из них.

Данными недостатками не обладает другой метод построения регрессионной модели, основанный на регуляризации по Тихонову, который называется эластичная сеть. Метод эластичная сеть основан на минимизации функционала

$$Q_{elnet}(\tilde{S}_t, \beta_0, \ldots, \beta_n) = \sum_{j=1}^{m}[y_j - \beta_0 - \sum_{i=1}^{n} \beta_i \breve{x}_{ji}^s]^2 + \gamma \sum_{i=1}^{n}[\beta_i^2 \theta + (1-\theta) \mid \beta_i \mid], \tag{29}$$

где $\theta \in [0, 1]$. Метод эластичная сеть включает в себя метод гребневая регрессия и Лассо как частные случаи.

Лекция 4

Задачи прогнозирования, Линейная машина, Теоретические методы оценки обобщающей способности,

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Использование формулы Байеса

Ранее было показано, что максимальную точность распознавания обеспечивает байесовское решающее правило, относящее распознаваемый объект, описываемый вектором $\boldsymbol{x}$ переменных (признаков) $X_1, \ldots, X_n$ к классу K_* , для которого условная вероятность $P(K_* \mid \boldsymbol{x})$ максимальна. Байесовские методы обучения основаны на аппроксимации условных вероятностей классов в точках признакового пространства с использованием формулы Байеса. Рассмотрим задачу распознавания классов $K_1, \ldots, K_L$. Формула Байеса позволяет рассчитать Условные вероятности классов в точке признакового пространства могут бфыть рассчитаны с использованием формулы Байеса. В случае, если переменные $X_1, \ldots, X_n$ являются дискретными формула Байеса может быть записана в виде:

$$P(K_i \mid \boldsymbol{x}) = \frac{P(\boldsymbol{x} \mid K_i)P(K_i)}{\sum_{i=1}^{L} P(K_i)P(\boldsymbol{x} \mid K_i)} \tag{1}$$

где $P(K_1), \ldots, P(K_L)$ - вероятность классов $K_1, \ldots, K_L$ безотносительно к признаковым описаниям (априорная вероятность).

Использование формулы Байеса

В качестве оценок априорных вероятностей

$$P(K_1), \ldots, P(K_L)$$

могут быть взяты доли объектов соответствующих классов в обучающей выборке. Условные вероятности $P(\boldsymbol{x} \mid K_1), \ldots, P(\boldsymbol{x} \mid K_L)$ могут оцениваться на основании сделанных предположений. Например, может быть использовано предположение о независимости переменных для каждого из классов. В последнем случае вероятность $P(\boldsymbol{x}_j \mid K_i)$ для вектора $\boldsymbol{x}_k = (x_{j1}, \ldots, x_{jn})$ может быть представлена в виде:

$$P(\boldsymbol{x}_j \mid K_i) = \prod_{i=1}^{n} P(X_j = x_{ki} \mid K_i). \tag{2}$$

Предположим, переменная X_j принимает значения из конечного множества $\widetilde{M}_j^i$ на объектах из класса K_i при $j = 1, \ldots, n$ и $i = 1, \ldots, L$. Предположим, что

$$\widetilde{M}_j^i = \{a_{ji}^1, \ldots, a_{ji}^{r(i,j).}\}$$

Наивный байесовский классификатор

Для того, чтобы воспользоваться формулой (2) достаточно знать вероятность выполнения равенства $X_j = a_{ji}^k$ для произвольного класса и произвольной переменной. Для оценки вероятности $P(X_j = a_{ji}^k | K_i)$ может использоваться доля объектов из $\widetilde{S}_t \bigcap K_i$, для которых $X_j = a_{ji}^k$. В случае, если переменные $X_1, \ldots, X_n$ являются непрерывными, формула Байеса может быть записана с использованием

$$P(K_i \mid \boldsymbol{x}) = \frac{p_i(\boldsymbol{x})P(K_i)}{\sum_{i=1}^{L} P(K_i)p_i(\boldsymbol{x})}, \tag{3}$$

где $p_1(\boldsymbol{x}), \ldots, p_L(\boldsymbol{x})$ - значения плотностей вероятностей классов $K_1, \ldots, K_L$ в пространстве $\mathbb{R}^n$.

лотности вероятностей

$$p_1(\boldsymbol{x}), \ldots, p_L(\boldsymbol{x})$$

также могут оцениваться исходя из предположения взаимной независимости переменных $X_1, \ldots, X_n$.

Наивный байесовский классификатор

В этом случае $p_i(\boldsymbol{x})$ может быть представлена в виде произведения одномерных плотностей

$$p_i(\boldsymbol{x}) = \prod_{j=1}^{n} p_{ji}(X_j),$$

где $p_{ji}(X_j)$ - плотность распределения переменной X_j для класса K_i. Плотности $p_{ji}(X_j)$ могут оцениваться в рамках предположения о типе распределения. Например, может использоваться гипотеза о нормальности распределений

$$p_{ji}(X_j) = \frac{1}{\sqrt{2\pi}D_{ji}} e^{\frac{-(X_j - M_{ji})^2}{2D_{ji}}},$$

где M_{ji},D_{ji} являются математическим ожиданием и дисперсией переменной X_j. Данне параметры легко оцениваются по $\widetilde{S}_t$.

Методы, основанные на формуле Байеса

Методы распознавания, основанные на использовании формулы Байеса в форме (1) и (3) и гипотезе о независимости переменных обычно называют наивными байесовскими классификаторами. Отметим, что знаменатели в правых частях формул (1) и (3) тождественны для всех классов. Поэтому при решении задач распознавания достаточно использовать только числители.

Аппроксимация плотности с помощь многомерного нормального распределения

При решении задач распознавания с помощью формулы Байеса в форме (3) могут использоваться плотности вероятности $p_1(\boldsymbol{x}), \ldots, p_L(\boldsymbol{x})$, в которых переменные $X_1, \ldots, X_n$ не обязательно являются независимыми. Чаще всего используется многомерное нормальное распределения. Плотность данного распределения в общем виде представляется выражением

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{\frac{n}{2}} \mid \Sigma \mid^{\frac{1}{2}}} exp[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})\Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu})^t], \tag{4}$$

где

$\boldsymbol{\mu}$ - математическое ожидание вектора признаков $\boldsymbol{x}$; Σ - матрица ковариаций признаков $X_1, \ldots, X_n$; $\mid \Sigma \mid$ -детерминант матрицы Σ.

Для построения распознающего алгоритма достаточно оценить вектора математических ожиданий $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_L$ и матрицы ковариаций $\Sigma_1, \ldots, \Sigma_L$ для классов $K_1, \ldots, K_L$, соответственно.

Аппроксимация плотности с помощь многомерного нормального распределения

Оценка вектора математических ожиданий $\boldsymbol{\mu}_i$ вычисляется как среднее значение векторов признаков по объектам обучающей выборки $\widetilde{S}_t$ из класса K_i :

$$\hat{\boldsymbol{\mu}}_i = \frac{1}{m_i} \sum_{s_j \in \widetilde{S}_t \bigcap K_i} \boldsymbol{x}_j$$

, где m_i - число объектов класса K_i в обучающей выборке. Элемент матрицы ковариаций для класса K_i вычисляется по формуле

$$\hat{\sigma}^i_{kk'} = \frac{1}{m_i} \sum_{s_j \in \widetilde{S}_t \bigcap K_i} (x_{jk} - \mu_{ik})(x_{jk'} - \mu_{ik'}),$$

где $x_{jk} - \mu_{ik}$ - k-я компонента вектора $\boldsymbol{\mu}_i$. Матрицу ковариации, состоящую из элементов $\hat{\sigma}^i_{kk'}$ обозначим $\hat{\Sigma}_i$. Очевидно, что согласно формуле Байеса максимум $P(K_i \mid \boldsymbol{x})$ достигается для тех же самых классов для которых максимально произведение $P(K_i)p_i(\boldsymbol{x})$.

Использование формулы Байеса. Многомерное нормальное распределение

Очевидно, что для байесовской классификации может использоваться также натуральный логарифм $\ln[P(K_i)p_i(\boldsymbol{x})]$ который согласно вышеизложенному может быть оценён выражением

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}\hat{\Sigma}_i^{-1}\boldsymbol{x}^t + \boldsymbol{w}_i\boldsymbol{x}^t + g_i^0,$$

где $\boldsymbol{w}_i = \hat{\boldsymbol{\mu}}_i\hat{\Sigma}_i^{-1}$ g_i^0 - не зависящее от $\boldsymbol{x}$ слагаемое:
ν_i - доля объектов класса K_i в обучающей выборке. Слагаемое g_i^0 имеет вид

$$g_i^0 = -\frac{1}{2}\hat{\boldsymbol{\mu}}_i\hat{\Sigma}_i^{-1}\hat{\boldsymbol{\mu}}_i^t - \frac{1}{2}\ln\left(\mid\hat{\Sigma}_i\mid\right) + \ln(\nu_i) - \frac{n}{2}\ln(2\pi).$$

Использование формулы Байеса. Многомерное нормальное распределение

Таким образом объект с признаковым описанием $\boldsymbol{x}$ будет отнесён построенной выше аппроксимацией байесовского классификатора к классу, для которого оценка $g_i(\boldsymbol{x})$ является максимальной. Следует отметить, что построенный классификатор в общем случае является квадратичным по признакам. Однако классификатор превращается в линейный, если оценки ковариационных матриц разных классов оказываются равными.

Использование формулы Байеса. Многомерное нормальное распределение

Рассмотрим вариант метода Линейный дискриминант Фишера (ЛДФ) для распознавания двух классов K_1 и K_2 . В основе метода лежит поиск в многомерном признаковом пространстве такого направления $\boldsymbol{w}$, чтобы средние значения проекции на него объектов обучающей выборки из классов K_1 и K_2 максимально различались. Проекцией произвольного вектора $\boldsymbol{x}$ на направление $\boldsymbol{w}$ является отношение

$$\frac{(\boldsymbol{w}\boldsymbol{x}^t)}{|\ \boldsymbol{w}\ |}.$$

В качестве меры различий проекций классов на используется функционал

$$\Phi(\boldsymbol{w}, \widetilde{S}_t) = \frac{(\hat{X}_{\boldsymbol{w}1} - \hat{X}_{\boldsymbol{w}2})^2}{\hat{d}_{\boldsymbol{w}1} + \hat{d}_{\boldsymbol{w}2}},$$

где

$$\hat{X}_{\boldsymbol{w}i} = \frac{1}{m_i} \sum_{s_j \in \widetilde{S}_t \bigcap K_i} \frac{(\boldsymbol{w}\boldsymbol{x}_j^t)}{\mid \boldsymbol{w} \mid}$$

- среднее значение проекции векторов переменных $X_1, \ldots, X_n$, описывающих объекты из класса K_i;

$$\hat{d}_{\boldsymbol{w}i} = \frac{1}{m_i} \sum_{s_j \in \widetilde{S}_t \bigcap K_i} [\frac{(\boldsymbol{w}\boldsymbol{x}_j^t)}{\mid \boldsymbol{w} \mid} - \hat{X}_{\boldsymbol{w}i}]^2$$

- дисперсия проекций векторов, описывающих объекты из класса $K_i, i \in \{1, 2\}$. Смысл функционала $\Phi(\boldsymbol{w}, \widetilde{S}_t)$ ясен из его структуры. Он является по сути квадратом отличия между средними значениями проекций классов на направление $\boldsymbol{w}$, нормированным на сумму внутриклассовых выборочных дисперсий.

Линейный дискриминант Фишера.

Можно показать, что $\Phi(\boldsymbol{w}, \widetilde{S}_t)$ достигает максимума при

$$\boldsymbol{w} = \hat{\Sigma}_{12}^{-1}(\hat{\boldsymbol{\mu}}_1^t - \hat{\boldsymbol{\mu}}_2^t), \tag{5}$$

где $\hat{\Sigma}_{12} = \hat{\Sigma}_1 + \hat{\Sigma}_2$. Таким образом оценка направления, оптимального для распознавания K_1 и K_2 может быть записана в виде (5) Распознавание нового объекта s_* по векторному описанию $\boldsymbol{x}_*$ производится по величине его проекции на направление $\boldsymbol{w}$:

$$\gamma(\boldsymbol{x}_*) = \frac{(\boldsymbol{w}, \boldsymbol{x}_*^t)}{\mid \boldsymbol{w} \mid}. \tag{6}$$

При этом используется простое пороговое правило: при $\gamma(\boldsymbol{x}_*) > b$ объект s_* относится к классу K_1 и s_* относится к классу K_2 в противном случае.

Линейный дискриминант Фишера.

Граничный параметр b подбирается по обучающей выборке таким образом, чтобы проекции объектов разных классов на оптимальное направление $\boldsymbol{w}$ оказались бы максимально разделёнными. Простой, но эффективной, стратегией является выбор в качестве порогового параметра b средней проекции объектов обучающей выборки на $\boldsymbol{w}$.
Метод ЛДФ легко обобщается на случай с несколькими классами. При этом исходная задача распознавания классов $K_1, \ldots, K_L$ сводится к последовательности задач с двумя классами K_1' и K_2' :

Зад. 1. Класс $K_1' = K_1$, класс $K_2' = \Omega \setminus K_1$

. . . . . .

Зад. L. Класс $K_1' = K_L$, класс $K_2' = \Omega \setminus K_L$

Для каждой из L задач ищется оптимальное направление. В результате получается набор из L направлений $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_L$.

Линейный дискриминант Фишера.

В результате получается набор из L направлений $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_L$. При распознавании нового объекта s_* по признаковому описанию $\boldsymbol{x}_*$ вычисляются проекции на $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_L$:

$$\gamma_1(\boldsymbol{x}_*) = \frac{(\boldsymbol{w}_1, \boldsymbol{x}_*^t)}{|\boldsymbol{w}_1|}, \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots, \gamma_L(\boldsymbol{x}_*) = \frac{(\boldsymbol{w}_L, \boldsymbol{x}_*^t)}{|\boldsymbol{w}_L|}.$$

Распознаваемый объект относится к тому классу, соответствующему максимальной величине проекции. Распознавание может производится также по величинам

$$[\gamma_1(\boldsymbol{x}_*) - b_1], \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots, [\gamma_L(\boldsymbol{x}_*) - b_L].$$

Логистическая регрессия.

Целью логистической регрессии является аппроксимация плотности условных вероятностей классов в точках признакового пространства. При этом аппроксимация производится с использованием логистической функции.

$$g(z) = \frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}}$$

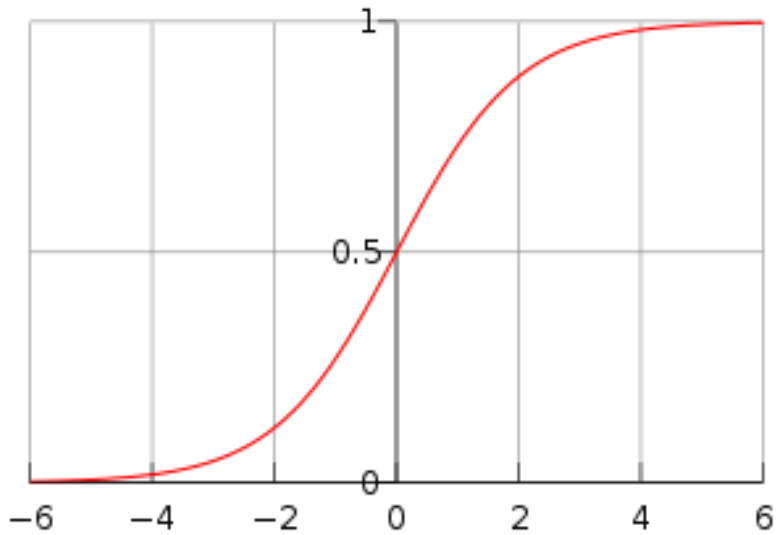


Рис 1. Логистическая функция.

Логистическая регрессия.

В методе логистическая регрессия связь условной вероятности класса K с прогностическими признаками осуществляются через переменную Z , которая задаётся как линейная комбинация признаков:

$$z = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n.$$

Условная вероятность K в точке векторного пространства $\boldsymbol{x}_* = (x_{1*}, \ldots, x_{n*})$ задаётся в виде

$$P(K \mid \boldsymbol{x}) = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_n X_n}}{1 + e^{\beta_0+\beta_1 X_1+\ldots+\beta_n X_n}} = \frac{1}{1 + e^{-\beta_0-\beta_1 X_1-\ldots-\beta_n X_n}} \qquad (7)$$

Логистическая регрессия.

Оценки регрессионных параметров $\beta_0, \beta_1, \ldots, \beta_n$ могут быть вычислены по обучающей выборке с помощью различных вариантов метода максимального правдоподобия. Предположим, что объекты обучающей выборки сосредоточены в точках признакового пространства из множества $\widetilde{\boldsymbol{x}} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r\}$. При этом распределение объектов обучающей выборка по точкам задаётся с помощью набора пар $\{(m_1, k_1), \ldots, (m_r, k_r)\}$, где m_i - общее число объектов в точке $\boldsymbol{x}_i$, k_i - число объектов класса K в точке $\boldsymbol{x}_i$. Вероятность данной конфигурации подчиняется распределению Бернулли. Введём обозначение $\varrho(\boldsymbol{x}) = P(K \mid \boldsymbol{x})$. Оценка вектора регрессионных параметров $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_n)$ может быть получена с помощью метода максимального правдоподобия. Функция правдоподобия может быть записана в виде

$$L(\boldsymbol{\beta}, \widetilde{\boldsymbol{x}}) = \prod_{j=1}^{r} C_{m_i}^{k_i} [\varrho(\boldsymbol{x})_j]^{k_j} [1 - \varrho(\boldsymbol{x})_j]^{(m_j - k_j)} \tag{8}$$

Принимая во внимание справедливость равенств

$$\frac{\varrho(\boldsymbol{x})}{1-\varrho(\boldsymbol{x})} = e^{\beta_0+\beta_1 X_1+\ldots+\beta_n X_n},$$

$$1-\varrho(\boldsymbol{x}) = \frac{1}{1+e^{\beta_0+\beta_1 X_1+\ldots+\beta_n X_n}},$$

приходим равенству

$$L(\boldsymbol{\beta},\widetilde{\boldsymbol{x}}) = \prod_{j=1}^{r} C_{m_i}^{k_i} e^{k_i\beta_0+\beta_1 x_{j1}+\ldots+\beta_n x_{jn}} \frac{1}{\left(1+e^{\beta_0+\beta_1 x_{j1}+\ldots+\beta_n x_j n}\right)^{m_i}} \quad (9)$$

Логистическая регрессия.

Поиск оптимального значения параметров удобнее производить, решая задачу максимизации логарифма функции правдоподобия, который в нашем случае принимает вид:

$$\ln[L(\boldsymbol{\beta}, \widetilde{\boldsymbol{x}})] = \sum_{j=1}^{r} \ln C_{m_j}^{k_j} + \sum_{j=1}^{r} [k_j(\beta_0 + \beta_1 x_{j1} + \ldots + \beta_n x_{jn})] +$$

$$+ \sum_{j=1}^{r} m_j \ln\left(\frac{1}{1 + e^{\beta_0 + \beta_1 x_{j1} + \ldots + \beta_n x_{jn}}}\right)$$

K-ближайших соседей.

Простым, но достаточно эффективным подходом к решению задач распознавания является метод k-ближайших соседей. Оценка условных вероятностей $P(K_i \mid \boldsymbol{x})$ ведётся по ближайшей окрестности V_k точки $\boldsymbol{x}$, содержащей k признаковых описаний объектов обучающей выборки. В качестве оценки выступает отношение $\frac{k_i}{k}$, где k_i - число признаковых описаний объектов обучающей выборки из K_i внутри V_k . Окрестность V_k задаётся с помощью функции расстояния $\rho(\boldsymbol{x}', \boldsymbol{x}'')$ заданной на декартовом произведении $\widetilde{X} \times \widetilde{X}$, где $\widetilde{X}$ - область допустимых значений признаковых описаний. В качестве функции расстояния может быть использована стандартная эвклидова метрика. То есть расстояние между двумя векторами $\boldsymbol{x}' = (x'_1, \ldots, x'_n)$ и $\boldsymbol{x}'' = (x''_1, \ldots, x''_n)$

$$\rho(\boldsymbol{x}', \boldsymbol{x}'') = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x'_i - x''_i)^2}.$$

K-ближайших соседей.

Для задач с бинарными признаками в качестве функции расстояния может быть использована метрика Хэмминга, равная числу совпадающих позиций в двух сравниваемых признаковых описаниях. Окрестность V_k ищется путём поиска в обучающей выборке $\widetilde{S}_t$ векторных описаний, ближайших в смысле выбранной функции расстояний, к описанию $\boldsymbol{x}_*$ распознаваемого объекта s_* . Единственным параметром, который может быть использован для настройки (обучения) алгоритмов в методе k–ближайших соседей является собственно само число ближайших соседей. Для оптимизации параметра k обычно используется метод, основанный на скользящем контроле. Оценка точности распознавания производится по обучающей выборке при различных k и выбирается значение данного параметра, при котором полученная точность максимальна.

Распознавание при заданной точности распознавания некоторых классов

Байесовский классификатор обеспечивает максимальную общую точность распознавания. Однако при решении конкретных практических задач потери, связанные с неправильной классификацией объектов, принадлежащих к одному из классов, значительно превышают потери, связанные с неправильной классификацией объектов других классов. Для оптимизации потерь необходимо использование методов распознавания с учётом предпочтительной точности распознавания для некоторых классов. Одним из возможных подходов является фиксирование порога для точности распознавания одного из классов. Оптимальное решающее правило в задаче распознавания с двумя классами K_1 и K_2 , обеспечивающее максимальную точность распознавания K_2 при фиксированной точности распознавания K_1 , описывается критерием Неймана-Пирсона.

Распознавание при заданной точности распознавания некоторых классов

Критерий Неймана-Пирсона Максимальная точность распознавания K_2 при точности распознавания K_1 равной α обеспечивается правилом: Объект с описанием $\boldsymbol{x}$ относится в класс K_1 , если

$$P(K_1 \mid \boldsymbol{x}) \geq \eta P(K_2 \mid \boldsymbol{x})$$

где параметр η определяется из условия

$$\int_{\Theta} P(K_1 \mid \boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x} = \alpha$$

$$\Theta = \{\boldsymbol{x} \mid P(K_1 \mid \boldsymbol{x}) \geq \eta P(K_2 \mid \boldsymbol{x})\}$$

$p(\boldsymbol{x})$ - плотность распределения $K_1 \bigcup K_2$ в точке $\boldsymbol{x}$. Критерий Неймана-Пирсона может быть использован, если известны плотности распределения распознаваемых классов. Плотности могут быть восстановлены в рамках Байесовских методов обучения на основе гипотез о виде распределений. ,

Распознавание при заданной точности распознавания некоторых классов

Критерий Неймана-Пирсона может быть использован, если известны плотности распределения распознаваемых классов. Плотности могут быть восстановлены в рамках Байесовских методов обучения на основе гипотез о виде распределений. Однако существуют эффективные средства регулирования точности распознавания при предпочтительности одного из классов, которые не требуют гипотез о виде распределения. Данные средства основаны на структуре распознающего алгоритма. Каждый алгоритм распознавания классов $K_1, \ldots, K_l$ может быть представлен как последовательное выполнение распознающего оператора $\mathbf{R}$ и решающего правила :

$$A = \mathbf{R} \otimes \mathbf{C}.$$

Оператор оценок вычисляет для распознаваемого объекта s вещественные оценки $\gamma_1, \ldots, \gamma_L$ за классы $K_1, \ldots, K_l$ соответственно.

Распознавание при заданной точности распознавания некоторых классов

Решающее правило производит отнесение объекта s по вектору оценок $\gamma_1, \ldots, \gamma_L$ к одному из классов. Распространённым решающим правилом является простая процедура, относящая объект в тот класс, оценка за который максимальна.
В случае распознавания двух классов K_1 и K_2 распознаваемый объект s будет отнесён к классу K_1 , если $\gamma_1(s) - \gamma_2(s) > 0$ и классу K_2 в противном случае. Назовём приведённое выше правило правилом $\boldsymbol{C}(0)$. Однако точность распознавания правила $\boldsymbol{C}(0)$ может оказаться слишком низкой для того, чтобы обеспечить требуемую величину потерь, связанных с неправильной классификацией объектов, на самом деле принадлежащих классу K_1 . Для достижения необходимой величины потерь может быть использовано пороговое решающее правило $\mathbf{C}(\delta)$.

Распознавание при заданной точности распознавания некоторых классов

Прравило $\mathbf{C}(\delta)$: распознаваемый объект s будет отнесён к классу K_1 , если $\gamma_1(s) - \gamma_2(s) > \delta$ и классу K_2 в противном случае. Обозначим через $p_{ci}(\delta, s)$ вероятность правильной классификации правилом $\mathbf{C}(\delta)$ объекта s , на самом деле принадлежащего K_i , $i \in \{1, 2\}$. При $\delta < 0$ $p_{c1}(\delta, s) \geq p_{c1}(0, s)$, но $p_{c2}(\delta, s) \leq p_{c2}(0, s)$. Уменьшая δ , мы увеличиваем $p_{c1}(\delta, s)$ и уменьшаем $p_{c2}(\delta, s)$. Напротив, увеличивая δ , мы уменьшаем $p_{c1}(\delta, s)$ и увеличиваем $p_{c2}(\delta, s)$. Зависимость между $p_{c1}(\delta, s)$ и $p_{c2}(\delta, s)$ может быть приближённо восстановлена по обучающей выборке $\widetilde{S}_t$, включающей описания объектов $\{s_1, \ldots, s_m\}$.

Распознавание при заданной точности распознавания некоторых классов

Пусть
$$\begin{pmatrix} \gamma_1(s_1) & \dots & \gamma_1(s_m) \\ \gamma_2(s_1) & \dots & \gamma_2(s_m) \end{pmatrix}$$
является матрицей оценок за классы K_1 и K_2 объектов из $\widetilde{S}_t$. Пусть

$$\gamma(s_1) = \gamma_1(s1) - \gamma_2(s_1), \dots, \gamma(s_m) = \gamma_1(s_m) - \gamma_2(s_m).$$

Предположим, что величины $[\gamma(s_1), \dots, \gamma(s_m)]$ принимают r значений $\Gamma_1, \dots, \Gamma_r$. Рассмотрим r пороговых решающих правил $[\boldsymbol{C}(\Gamma_1), \dots, \boldsymbol{C}(\Gamma_r)]$ Для каждого из правил $\mathbf{C}(\Gamma_i)$ обозначим через $\nu_{c1}(\Gamma_i)$ долю K_1 среди объектов обучающей выборки, удовлетворяющих условию $\gamma(s_*) \geq \Gamma_i$, а через $\nu_{c2}(\Gamma_i)$ обозначим долю K_2 среди объектов обучающей выборки, удовлетворяющих условию $\gamma(s_*) < \Gamma_i$.

Распознавание при заданной точности распознавания некоторых классов. ROC анализ.

Отобразим результаты расчётов

$$\{[\nu_{c1}(\Gamma_1), \nu_{c2}(\Gamma_1)] \ldots, [\nu_{c1}(\Gamma_r), \nu_{c2}(\Gamma_r)]\}$$

как точки на в декартовой системе координат. Соединив точки отрезками прямых, получим ломаную линию (l), соединяющую точки (1,0) и (0,1). Данная линия графически отображает аппроксимацию по обучающей выборке взаимозависимости между $p_{c1}(\delta, s)$ и $p_{c2}(\delta, s)$ при всевозможных значениях δ . Соответствующий пример представлен на рисунке 2.

Распознавание при заданной точности распознавания некоторых классов. ROC анализ.

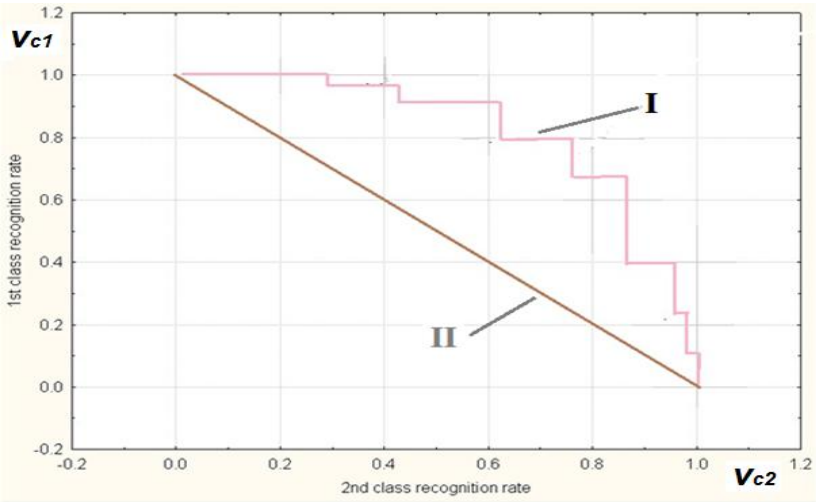


Рис.2

Взаимозависимость между ν_{c1} и ν_{c2} наиболее полно оценивает эффективность распознающего оператора $\mathbf{R}$. Отметим, что ν_{c1} постепенно убывает по мере роста ν_{c2} . .

Распознавание при заданной точности распознавания некоторых классов. ROC анализ.

Однако сохранение высокого значения ν_{c1} при высоких значениях ν_{c2} соответствует существованию решающего правила, при котором точность распознавания обоих классов высока. Таким образом эффективному распознающему оператору соответствует близость линии I к прямой, связывающей точки (0,1) и (1,1). То есть наиболее высокой эффективности соответствует максимально большая площадь под линией I. Отсутствию распознающей способности соответствует близость к прямой II, связывающей точки (0, 1) и (1,0).На рисунке 3 сравниваются линии, характеризующие эффективность распознающих операторов, принадлежащих к трём методам распознавания, при решении задачи распознавания двух видов аутизма по психометрическим показателям .

Распознавание при заданной точности распознавания некоторых классов. ROC анализ.

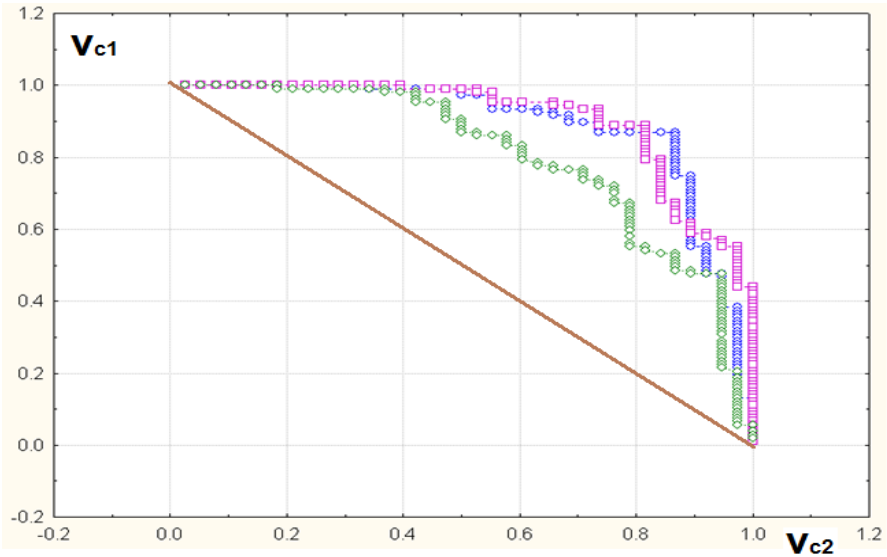


Рис.3

- - линейный дискриминант Фишера;
- - метод опорных вектор;
- - статистически взвешенные синдромы.

ROC анализ.

Методы распознавания используются при решении многих задач идентификации объектов, представляющих важность для пользователя. Эффективность идентификации для таких задач удобно описывать в терминах: «Чувствительность» - доля правильно распознанных объектов целевого класса «Ложная тревога» - доля объектов ошибочно отнесённых в целевой класс. Пример кривой, связывающей параметры «Чувствительность» и «Ложная тревога» представлен на рисунке 4. Анализ, основанный на построении и анализе линий, связывающих параметры «Чувствительность» и «Ложная тревога» принято называть анализом Receiver Operating Characteristic или ROC-анализом. Линии, связывающих параметры «Чувствительность» и «Ложная тревога» принято называть ROC-кривыми.

ROC анализ.

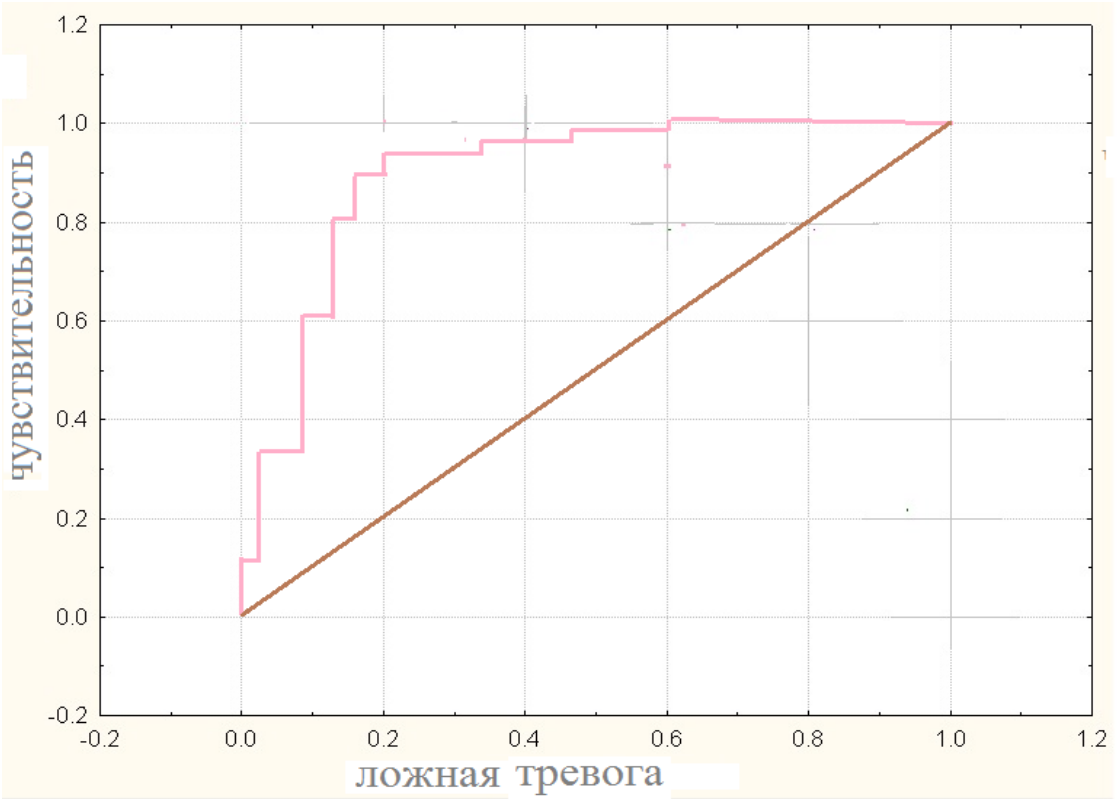


Рис.4. Пример ROC кривой

Лекция 5

Прицип частичной прецедентности, тестовый алгоритм, модель ABO

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Принцип частичной прецедентности. Тестовый алгоритм.

Существует ряд методов распознавания, основанных на принципе частичной прецедентности. Данный принцип подразумевает поиск по обучающей выборке фрагментов описаний, позволяющих разделить распознаваемые классы $K_1, \ldots, K_L$. Распознаваемый объект оценивается по совокупности найденных фрагментов. Одной из первых реализаций принципа частичной прецедентности является тестовый алгоритм, предложенный в 1966 году. Данный алгоритм основан на понятии тупикового теста. Исходный вариант тестового алгоритма предназначен для распознавания объектов, описываемых с помощью бинарных или категориальных признаков $X_1, \ldots, X_n$. Иными словами $X_i \in \{a_1^i, \ldots, a_{k_i}^i\}$, , $i = 1, 2, \ldots, n$.

Тестовый алгоритм

Пусть обучающая выборка $\widetilde{S}_t$, содержит объекты из классов $K_1, \ldots, K_L$. При этом общее число объектов равно m. Выборке $\widetilde{S}_t$ ставится в соответствие таблица $\mathbb{T}_{nmL}$, j-ой строкой которой являются значения признаков для объекта s_j.

Определение 1. Тестом таблицы $\mathbb{T}_{nmL}$ называется совокупность столбцов $i_1, \ldots, i_r$ таких, что после удаления из $\mathbb{T}_{nmL}$ всех столбцов, за исключением $i_1, \ldots, i_r$, в полученной таблице все пары строк, соответствующие разным классам различны хотя бы по одному признаку

Определение 2. Тест $T = \{i_1, \ldots, i_r\}$ называется тупиковым, если никакое его отличное от T подмножество (собственное подмножество) тестом не является.

На этапе обучения ищется множество всевозможных тупиковых тестов $\widetilde{T}(\widetilde{S}_t)$ для таблицы $\mathbb{T}_{nmL}$. Предположим что нам требуется распознать объект s_* с векторным описанием $(x_{*1}, \ldots, x_{*n})$.

Тупиковые тесты

Выделим в векторном описании фрагмент $(x_{*i_1}, \ldots, x_{*i_r})$, соответствующий тесту

$$T = \{i_1, \ldots, i_r\}, T \in \widetilde{T}(\widetilde{S}_t).$$

Фрагмент $(x_{*i_1}, \ldots, x_{*i_r})$ сравнивается с множеством фрагментов строк таблицы $\mathbb{T}_{nmL}$, соответствующих классу K_l : $\{(x^t_{ji_1}, \ldots, x^t_{ji_r}) \mid s_j \in K_l\}$. В случаях, когда выполняются равенства

$$x_{*i_1} = x^t_{ji_1}, \ldots, x_{*i_r} = x^t_{ji_r}$$

фиксируем полное совпадение распознаваемого объекта s_* с объетом $s_j \in K_l$ тесту T. Обозначим число полных совпадений через $G_l(\mathbb{T}_{nmL}, s_*, T)$. Оценка объекта s_* за класс K_l вычисляется по формуле:

$$\gamma(s_*) = \frac{1}{m_l} \sum_{T \in \widetilde{T}(\widetilde{S}_t)} G_l(\mathbb{T}_{nmL}, s_*, T),$$

где m_l - число
объектов обучающей выборки из класса K_l .

Тупиковые тесты

Классификация объекта s_* может производится по вектору оценок $[\gamma_1(s_*), \ldots, \gamma_L(s_*)]$ с помощью стандартного решающего правила, т.е. объект относится в тот класс, оценка за который максимальна. Задача нахождения множества всех тупиковых тестов таблицы $\mathbb{T}_{nmL}$ сводится к задаче поиска всех тупиковых покрытий матрицы сравнений $\mathbb{C}_{nmL}$, которая строится по матрице $\mathbb{T}_{nmL}$. Каждой паре классов K_l и K'_l в матрице $\mathbb{C}_{nmL}$ сопоставлена подматрица $\mathbb{C}^{ll'}_{nmL}$, состоящая из $m_l m_{l'}$ строк. Пусть $(c^{ll'}_{f1}, \ldots, c^{ll'}_{fn})$ строка матрицы $\mathbb{C}^{ll'}_{nmL}$ соответствует сравнению описаний объекта $\boldsymbol{x}_g$ из класса K_l и описание объекта $\boldsymbol{x}_{g'}$ из класса $K_{l'}$. Элемент $c^{ll'}_{fi} = 0$, если $x_{gi} = x_{g'i}$, и $c^{ll'}_{fi} = 1$, если $x_{gi} \neq x_{g'i}$. Таким образом $\mathbb{C}_{nmL}$ имеет размерность $M \times n$, где$\sum_{l=1}^{L} \sum_{l'=1}^{l-1} m_l m_{l'}$.

Методы поиска тупиковых тестов. Задача о покрытии.

Мы будем говорить, что столбец с номером j матрицы $\mathbb{C}_{nmL}$ покрывает строку $(c_{f1}, \ldots, c_{fn})$, если $c_{fj} = 1$. Набор столбцов $\{j_1, \ldots, j_r\}$ образует покрытие матрицы $\mathbb{C}_{nmL}$, если при любом $f \in \{1, \ldots, M\}$ существует такое $j' \in \{j_1, \ldots, j_r\}$, что $c_{fj'} = 1$. Покрытие $\widetilde{J}$ называется тупиковым, если его произвольное собственное подмножество, покрытием не является. Очевидно, что для произвольного набора столбцов обладание свойством тупикового набора для $\mathbb{T}_{nmL}$ эквивалентно обладанию свойством тупикового покрытия для $\mathbb{C}_{nmL}$. Таким образом задача о поиске всевозможных тупиковых тестов сводится к известной задаче о поиске всевозможных тупиковых покрытий. Нахождение всех тупиковых тестов является сложной комбинаторной задачей. Однако эффективные алгоритмы поиска разработаны для некоторых типов таблиц. При решении практических задач эффективен подход, основанный на вычислении только части тупиковых тестов.

Предстаительные наборы

Другим известным классом алгоритмов распознавания, основанным на принципе частичной прецедентности, являются алгоритмы типа КОРА. В отличие от тестового алгоритма, где в качестве информативных элементов используются несжимаемые наборы признаков – тупиковые тесты, в алгоритмах типа КОРА в качестве информативных элементов используются несжимаемые фрагменты описаний эталонных объектов обучающей выборки.

Определение 3. Пусть $(x_{v1}, \ldots, x_{v_n})$ - признаковое описание объекта $s_v \in \widetilde{S}_t \bigcap K_l$. Набор $(x_{vj_1}, \ldots, x_{vj_r})$ называется представительным набором для класса K_l при выполнении следующего условия. Пусть $(x_{uj_1}, \ldots, x_{uj_r})$ произвольная строка таблицы $\mathbb{T}_{nmL}$, которая соответствует объекту s_u, не принадлежащему $\widetilde{S}_t \bigcap K_l$. Тогда существует такое $j' \in \{j_1, \ldots, j_r\}$, что $x_{uj'} \neq x_{vj'}$.

Представительные наборы

Определение 4. Представительный набор называется тупиковым, если никакое его собственное подмножество представительным набором не является.

На этапе обучения для каждого из классов $K_1, \ldots, k_L$ по таблице $\mathbb{T}_{nmL}$ ищется множество всевозможных тупиковых представительных наборов. Пусть $\widetilde{V}_l$ - множество всевозможных представительных наборов для класса K_l.

Предположим, что нам требуется распознать объект s_* с описанием $(x_{*1}, \ldots, x_{*n})$. Пусть $\boldsymbol{v} = (x_{vj_1}, \ldots, x_{vj_r})$ - представительный набором Функция $B(s_*, \boldsymbol{v})$ равна 1, при выполнении всех неравенств $(x_{*j_1} = x_{vj_1}, \ldots, x_{*j_r} = x_{vj_r})$ и $B(s_*, \boldsymbol{v})$ равна 0 в противном случае. Оценка s_* за класс K_l вычисляется по формуле

$$\Gamma(s_*, K_l) = \frac{1}{\mid \widetilde{V}_l \mid} \sum_{\boldsymbol{v} \in \widetilde{V}_l} B(s_*, \boldsymbol{v})$$

Представительные наборы

Для нахождения тупиковых представительных наборов для класса K_l , содержащихся в эталонном описании $\boldsymbol{x}_v$ объекта s_v формируются матрица сравнения $\mathbb{C}^{lv}_{nmL}$ со всеми описаниями других классов таблицы $\mathbb{T}_{nmL}$. Пусть строка $(c^{lv}_{f1}, \ldots, c^{lv}_{fn})$ матрицы $\mathbb{C}^{lv}_{nmL}$ соответствует сравнению $\boldsymbol{x}_v$ с описанием $\boldsymbol{x}_g$ объекта s_g, не принадлежащему $K_l \bigcap \widetilde{S}_t$. Элемент $c^{lv}_{f1} = 0$, если $x_{vj} = x_{gj}$, и $c^{lv}_{f1} = 1$, если $x_{vj} \neq x_{gj}$. Таким образом матрица $\mathbb{C}^{lv}_{nmL}$ имеет размер $(m - m_l)n$. Тупиковые покрытия матриц сравнения $\mathbb{C}^{lv}_{nmL}$ определяют тупиковые представительные наборы, являющиеся фрагментами описания $\boldsymbol{x}_v$. Полное множество представительных наборов для класса K_l является объединением множеств представительных наборов, найденных для описаний всех объектов обучающей выборки из K_l, Таким образом задача поиска всех представительных наборов сводится к решению m_l задач поиска тупиковых покрытий для матриц сравнения размера $(m - m_l)n$.

Представительные наборы

Первоначальные варианты тестового алгоритма и алгоритма типа КОРА были разработаны для бинарных или категориальных переменных. Они не могут быть напрямую использованы в задачах с признаками, принимающими значения из интервалов вещественной оси. Для того, чтобы обеспечить возможность работы с подобной информацией могут быть использованы два подхода.
Первый подход основан на разбиении области возможных значений каждого веществennозначного признака на k связных подмножеств (интервалов, полуинтервалов, отрезков). Значению признака, принадлежащего j-ому элементу разбиения присваивается значение j. Разбиение оптимизируется с целью достижения максимального разделения классов. Выбирается такое число элементов разбиения k, при котором достигается максимальная точность распознавания.

Веществознозначная информация

Другой подход основан на модификации понятий теста и представительного набора с использованием пороговых параметров $(\varepsilon_1, \ldots, \varepsilon_n)$, которые задаются для признаков $X_1, \ldots, X_n$.

Определение 5. Тестом таблицы $\mathbb{T}_{nmL}$ называется совокупность столбцов $i_1, \ldots, i_r$ таких, что после удаления из $\mathbb{T}_{nmL}$ всех столбцов, за исключением столбцов $i_1, \ldots, i_r$ в полученной таблице $\mathbb{T}_{rmL}$ для всех пар строк, соответствующих разным классам абсолютная величина различий хотя бы по одному признаку X_j превышает ε_j .

Аналогичным образом вводится модифицированное определение представительного набора.

Определение 6. Пусть $(x_{v1}, \ldots, x_{vn})$- признаковое описание объекта $s_v \in \widetilde{S}_t \bigcap K_l$. Набор $(x_{vj_1}, \ldots, x_{vj_r})$ называется представительным набором для класса K_l при выполнении следующего условия. Пусть $(x_{uj_1}, \ldots, x_{uj_r})$ произвольная строка таблицы $\mathbb{T}_{nmL}$ соответствующая объекту s_u, не принадлежащему $\widetilde{S}_t \bigcap K_l$. Тогда существует такое $j' \in \{j_1, \ldots, j_r\}$, что $\mid x_{uj'} - x_{vj'} \mid > \varepsilon_{j'}$.

Главным требованием при выборе ε-порогов является достижение максимальной отделимости объектов разных классов при сохранении сходства внутри классов. Поиск тупиковых тестов и тупиковых представительных наборов при модифицированных определениях аналогичен их поиску в первоначальных вариантах методов.

Алгоритмы вычисления оценок

Тестовый алгоритм и алгоритм с представительными наборами являются частью более общей конструкции, основанной на принципе частичной прецедентности и носящей название алгоритмов вычисления оценок. Существует много вариантов моделей данного типа. Причём конкретный вид модели определяется выбранными способами задания различных её элементов. Рассмотрим основные составляющие модели.

Задание системы опорных множеств. Под опорными множествами модели ABO понимается наборы признаков, по которым осуществляется сравнение распознаваемых и эталонных объектов. Примером системы опорных множеств является множество тупиковых тестов. Система опорных множеств Ω_A некоторого алгоритма A может задаваться через систему подмножеств множества $\{1, \ldots, n\}$ или через систему характеристических бинарных векторов.

Алгоритмы вычисления оценок

Каждому подмножеству $\{1, \ldots, n\}$ может быть сопоставлен бинарный вектор размерности n . Пусть $\{i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\}$. . Тогда$\{i_1, \ldots, i_k\}$ сопоставляется вектор $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$, все компоненты которого равны 0 кроме равных 1 компонент $(\omega_{i_1}, \ldots, \omega_{i_k})$. . Теоретические исследования свойств тупиковых тестов для случайных бинарных таблиц показали, что характеристические векторы для почти всех тупиковых тестов имеют асимптотически (при неограниченном возрастании размерности таблицы обучения) одну и ту же длину. Данный результат является обоснованием выбора в качестве системы опорных векторов всевозможных наборов, включающих фиксированное число признаков k или

$$\Omega_A = \{\boldsymbol{\omega} :| \omega |= k\}$$

Оптимальное значение k находится в процессе обучения или задаётся экспертом. .

Другой часто используемой системе опорных множеств соответствует множество всех подмножеств $\{1, \ldots, n\}$ за исключением пустого множества. Иными словами в систему опорных множеств входит произвольный набор признаков или $\Omega = \{\boldsymbol{\omega}\} \backslash \omega_0$, где ω_0 - вектор, все компоненты которого равны 0.

Задание функции близости. Пусть опорное множество $\{i_1, \ldots, i_k\}$ соответствует характеристическому вектору $\boldsymbol{\omega}$. Фрагмент $(x_{\mu i_1}, \ldots, x_{\mu i_k})$ описания $(x_{\mu 1}, \ldots, x_{\mu n})$ объекта s_μ называется $\boldsymbol{\omega}$-частью объекта s_μ.

Под функцией близости $B_{\boldsymbol{\omega}}(s_\mu, s_\nu)$ понимается функция от соответствующих $\boldsymbol{\omega}$-частей сравниваемых объектов, принимающая значения 1 (объекты близки) или 0 (объекты удалены). Функции близости обычно задаются с помощью порговых параметров $(\varepsilon_1, \ldots, \varepsilon_n)$, характеризующих близость объектов по отдельным признакам.

Пример функций близости.

1) $B_{\boldsymbol{\omega}}(s_\mu, s_\nu) = 1$, если при произвольном $i \in \{1, \ldots, n\}$, при котором $\omega_i = 1$, всегда выполняется неравенство

$$| x_{\mu i} - x_{\nu i} | < \varepsilon_i.$$

$B_{\boldsymbol{\omega}}(s_\mu, s_\nu) = 0$,, если существует такое $i' \in \{1, \ldots, n\}$, что одновременно $\omega'_i = 1$ и $| x_{\mu i'} - x_{\nu i} | > \varepsilon'_i$.

2) Пусть ε - скалярный пороговый параметр. Функция $B_{\boldsymbol{\omega}}(s_\mu, s_\nu) = 1$, если выполняется неравенство

$$[\sum_{i=1}^{n} \omega_i \mid x_{\mu i} - x_{\nu i} \mid] < \varepsilon.$$

В противном случае $B_{\boldsymbol{\omega}}(s_\mu, s_\nu) = 0$.

Алгоритмы вычисления оценок (АВО)

Важным элементом АВО является оценка близости распознаваемого объекта s_* к эталону s_μ по заданной $\boldsymbol{\omega}$- части. Данная оценка близости, которая будет обозначаться $\Gamma_{\boldsymbol{\omega}}(s_*, s_\mu)$, формируется на основе введённых ранее функций близости и, возможно, дополнительных параметров.

$$A)\Gamma_{\boldsymbol{\omega}}(s_*, s_\mu) = B_{\boldsymbol{\omega}}(s_*, s_\mu).$$
$$B)\Gamma_{\boldsymbol{\omega}}(s_*, s_\mu) = p_{\boldsymbol{\omega}} B_{\boldsymbol{\omega}}(s_*, s_\mu),$$

где $p_{\boldsymbol{\omega}}$- параметр, характеризующий информативность опорного множества с характеристическим вектором $\boldsymbol{\omega}$.

$$C)\Gamma_{\boldsymbol{\omega}}(s_*, s_\mu) = \gamma_\mu(\sum_{j=1}^{n} p_j \omega_j) B_{\boldsymbol{\omega}}(s_*, s_\mu),$$

где γ_μ - параметр, характеризующий информативность эталонного объекта s_μ , параметры $p_1, \ldots, p_n$ характеризуют информативность отдельных признаков.

Алгоритмы вычисления оценок (ABO)

Оценка объекта s_* за класс K_l при фиксированном $\boldsymbol{\omega}$. Оценка объекта s_* за класс K_l при фиксированном характеристическом векторе $\boldsymbol{\omega}$ может вычисляться как среднее значение близости s_* к эталонным объектам из класса K_l

$$\Gamma_{\boldsymbol{\omega}}^{l}(s_*) = \frac{1}{m_l} \sum_{s_\mu \in K_l} \Gamma_{\boldsymbol{\omega}}(s_*, s_\mu).$$

Общая оценка s_* за класс K_l вычисляется как сумма оценок $\Gamma_{\boldsymbol{\omega}}^{l}(s_*)$ по опорным множествам из системы Ω_A:

$$\Gamma^{l}(s_*) = \sum_{\omega \in \Omega_A} \Gamma_{\boldsymbol{\omega}}^{l}(s_*). \tag{1}$$

Наряду с формулой (1) используется формула

$$\Gamma^{l}(s_*) = \nu_l \sum_{\omega \in \Omega_A} \Gamma_{\boldsymbol{\omega}}^{l}(s_*). \tag{2}$$

Алгоритмы вычисления оценок (ABO)

Использование взвешивающих параметров $\nu_1, \ldots, \nu_L$ позволяет регулировать доли правильно распознанных объектов $K_1, \ldots, K_L$. Прямое вычисление оценок за классы по формулам (1) и (2) в случаях, когда в качестве систем опорных множеств используются наборы с фиксированным числом признаков или всевозможные наборы признаков, оказывается практически невозможным при сколь либо высокой размерности признакового пространства из-за необходимости вычисления огромного числа значений функций близости. Однако при равенстве весов всех признаков существуют эффективные формулы для вычисления оценок по формуле (1). Предположим, что оценки близости распознаваемого объекта s_* к эталону s_μ по заданной $\boldsymbol{\omega}$ - части вычисляются по формуле (A). Тогда оценка по формуле (1) принимает вид

$$\Gamma^l(s_*) = \sum_{\boldsymbol{\omega} \in \Omega_A} \sum_{s_\mu \in K_l} B_{\boldsymbol{\omega}(s_*, s_\mu)}$$

Рассмотрим сумму $\sum_{\boldsymbol{\omega}\in\Omega_A} B_{\boldsymbol{\omega}}(s_*, s_\mu)$. Предположим, что общее число признаков, по которым объект s_* близок к объекту s_μ равно $\boldsymbol{d}(s_*, s_\mu)$. Иными словами $\boldsymbol{d}(s_*, s_\mu) =\mid D(s_*, s_\mu) \mid$, где $D(s_*, s_\mu) = \{i :\mid x_{*i} - x_{\mu i} \mid< \varepsilon_i\}$. Очевидно функция близости $B_{\boldsymbol{\omega}}(s_*, s_\mu) = 1$ тогда и только тогда, когда опорное множество, задаваемое характеристическим вектором $\boldsymbol{\omega}$, полностью входит в множество $D(s_*, s_\mu)$. Во всех остальных случаях $B_{\boldsymbol{\omega}}(s_*, s_\mu) = 0$.

Алгоритмы вычисления оценок (ABO)

Предположим, что система опорных множеств удовлетворяет условию $\Omega_A = \{\boldsymbol{\omega} :| \boldsymbol{\omega} :|= k\}$. Очевидно, что число опорных множеств в Ω_A , удовлетворяющих условию $B_{\boldsymbol{\omega}}(s_*, s_\mu) = 1$, равно $C^k_{\boldsymbol{d}(s_*,s_\mu)}$. Откуда следует, что $\sum_{\boldsymbol{\omega}\in\Omega_A} B_{\boldsymbol{\omega}}(s_*, s_\mu) = C^k_{\boldsymbol{d}(s_*,s_\mu)}$. Следовательно оценка по формуле (1) может быть записана в виде

$$\Gamma^l(s_*) = \frac{1}{m_l} \sum_{s_\mu\in K_l} \gamma_\mu C^k_{\boldsymbol{d}(s_*,s_\mu)} \tag{3}$$

Предположим, что система Ω_A включает в себя всевозможные опорные множества. В этом случае число опорных множеств в Ω_A , удовлетворяющих условию $B_{\boldsymbol{\omega}}(s_*, s_\mu) = 1$, равно $2^{\boldsymbol{d}(s_*,s_\mu)} - 1$. Следовательно оценка по формуле (1) может быть записана в виде

$$\Gamma^l(s_*) = \frac{1}{m_l} \sum_{s_\mu\in K_l} \gamma_\mu [2^{\boldsymbol{d}(s_*,s_\mu)} - 1].$$

Обучение АВО

Для обучения алгоритмов АВО в общем случае может быть использован тот же самый подход, который используется для обучения в методе «Линейная машина». Предположим, что решается задача обучения алгоритмов для распознавания объектов , принадлежащих классам $K_1, \ldots, K_L$ При правильного распознавания объекта $s_i \in \widetilde{S}_t \bigcap K_l$ должен выполняться блок неравенств

$$\Gamma^l(s_i) > \Gamma^{l\prime}(s_i), \tag{4}$$

где $l' \in \{1, \ldots, L\} \setminus \{l\}$. При использовании для вычисления оценок формулы (3) блок (4) приводится к виду

$$\frac{1}{m_l} \sum_{s_\mu \in K_l} \gamma_\mu C^k_{\boldsymbol{d}(s_*, s_\mu)} > \frac{1}{m_l} \sum_{s_\nu \in K_{l'}} \gamma_\mu C^k_{\boldsymbol{d}(s_*, s_\nu)}$$

Обучение АВО

Обучение сводится к поиску такого набора весовых коэффициентов $\gamma_1, \ldots, \gamma_m$ при которых выполняется по возможности максимальное число блоков неравенств вида (4). Поиск максимальной оптимальных коэффициентов может производиться с использованием эвристического релаксационного метода, аналогичного тому, что был использован при обучении алгоритма «Линейная машина».

Лекция 6

Нейросетевые методы, перцептрон Розенблатта, многослойный перцептрон

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Нейросетевые методы

В основе нейросетевых методов лежит попытка компьютерного моделирования процессов обучения, используемых в живых организмах. Когнитивные способности живых существ связаны с функционированием сетей связанных между собой биологических нейронов – клеток нервной системы. Для моделирования биологических нейросетей используются сети, узлами которых являются искусственные нейроны (т.е. математические модели нейронов), Можно выделить три типа искусственных нейронов: нейроны-рецепторы, внутренние нейроны и реагирующие нейроны. Каждый внутренний или реагирующий нейрон имеет множество входных связей, по которым поступают сигналы от рецепторов или других внутренних нейронов. Пример модели внутреннего или реагирующего нейрона представлен на рисунке 1.

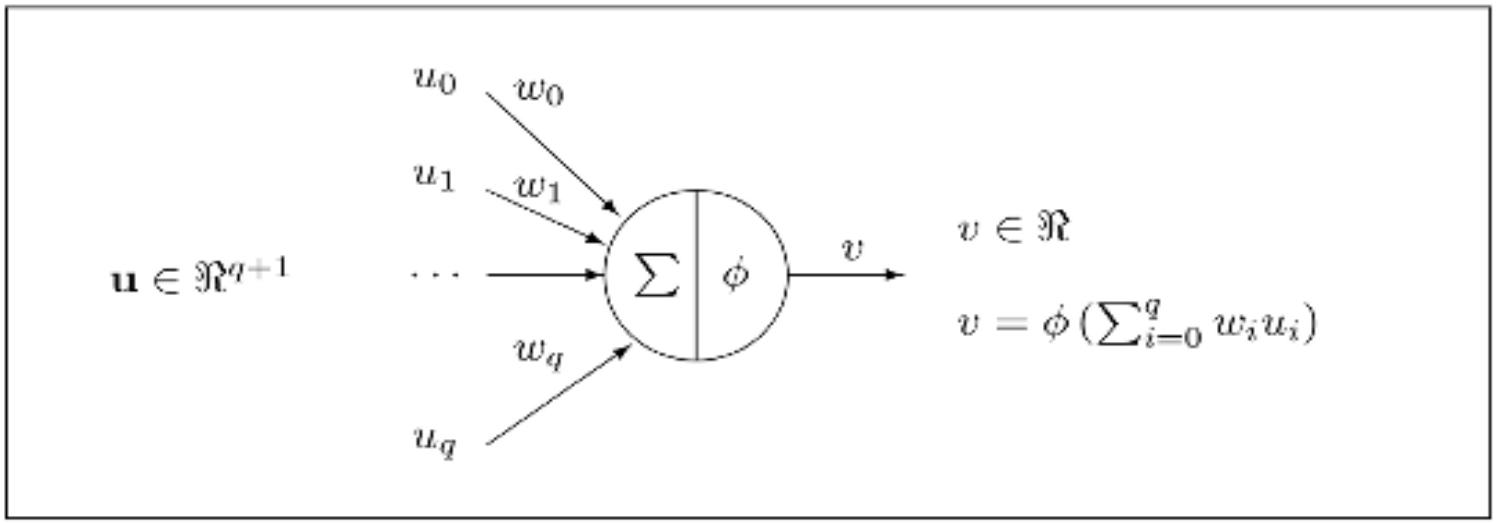


Рис.1. Модель внутреннего или реагирующего нейрона.

Представленный на рисунке 1 нейрон имеет r внешних связей, по которым на него поступают входные сигналы $u_1, \ldots, u_r$. Поступившие сигналы суммируются с весами $w_1, \ldots, w_r$. На выходе нейрона вырабатывается сигнал $\Phi(z)$, где $z = w_0 + \sum_{i=1}^{r} w_i u_i$, w_0 - параметр сдвига.Может быть использована также форма записи $z = \sum_{i=0}^{r} w_i u_i$, где фиктивный «сигнал» u_0 тождественно равен 1.

Функцию $\Phi(z)$ обычно называют активационной функцией. Могут использоваться различные виды активационных функций, включая

- пороговую функцию, задаваемую с помощью пороговой величины b: $\Phi(z) = 1$ при $z \geq b$, $\Phi(z) = 0$ при $z < b$;
- - сигмоидная функция $\Phi(z) = \frac{1}{1+e^{-az}}$, где a-вещественная константа;
- гиперболический тангенс;
- тождественное преобразование $\Phi(z) = z$.

Методы, основанные на использовании искусственных нейронов могут быть использованы для решения самых разнообразных задач распознавания. При этом сигналы, поступающие на вход перцептрона, интерпретируются как входные признаки $X_1, \ldots, X_n$.

Перцептрон Розенблатта

Первой нейросетевой моделью стал перцептрон Розенблатта, предложенный в 1957 году. В данной модели используется единственный реагирующий нейрон. Модель, реализующая линейную разделяющую функцию в пространстве входных сигналов, может быть использована для решении задач распознавания с двумя классами, помеченными метками 1 или -1. В качестве активационной функции используется пороговая функция: $\Phi(z) = 1$ при $z \geq 0$, $\Phi(z) = -1$ при $z < 0$. Особенностью модели Розенблатта является очень простая, но вместе с тем эффективная, процедура обучения, вычисляющая значения весовых коэффициентов $(w_0, \ldots, w_n)$. Настройка параметров производится по обучающим выборкам, совершенно аналогичных тем, которые используются для обучения статистических алгоритмов. На первом этапе производится преобразование векторов сигналов (признаковых описаний) для объектов обучающей выборки. В набор исходных признаков добавляется тождественно равная 1 нулевая компонента. Затем вектора описаний из класса K_2 умножаются на -1. Вектора описаний из класса K_1 не изменяются.

Перцептрон Розенблатта

Процедура обучения перцептрона. . Нулевое приближение вектора весовых коэффициентов $(w_0^{(0)}, \ldots, w_n^{(0)})$ выбирается случайным образом Преобразованные описания объектов обучающей выборки $\widetilde{S}_t$ последовательно подаются на вход перцептрона. В случае если описание $\boldsymbol{x}^{(k)}$, поданное на k-ом шаге классифицируется неправильно, то происходит коррекция по правилу $w^{(k+1)} = w^{(k)} + \boldsymbol{x}$. В случае правильной классификации $w^{(k+1)} = w^{(k)}$. Отметим, что правильной классификации всегда соответствует выполнение равенства $(w^{(k)}, \boldsymbol{x}^{(\boldsymbol{k})}) \geq 0$, а неправильной $(w^{(k)}, \boldsymbol{x}^{(\boldsymbol{k})}) < 0$. Процедура повторяется до тех пор, пока не будет выполнено одно из следующих условий:

- - достигается полное разделение объектов из классов K_1 и K_2;
- - повторение подряд заранее заданного числа итераций не приводит к улучшению разделения;
- - оказывается исчерпанным заранее заданный лимит итераций.

Перцептрон Розенблатта. Многослойный перцептрон.

Для описанной процедуры обучения справедлива следующая теорема.
Теорема. **В случае, если описания объектов обучающей выборки линейно разделимы в пространстве признаковых описаний, то процедура обучения перцептрона построит линейную гиперплоскость разделяющую объекты двух классов за конечное число шагов.**
Отсутствие линейной разделимости двух классов приводит к бесконечному зацикливанию gроцедуры обучения перцептрона.
Существенно более высокой аппроксимирующей способностью обладают нейросетевые методы распознавания, задаваемые комбинациями связанных между собой нейронов. Таким методом является Многослойный перцептрон.

Многослойный перцептрон

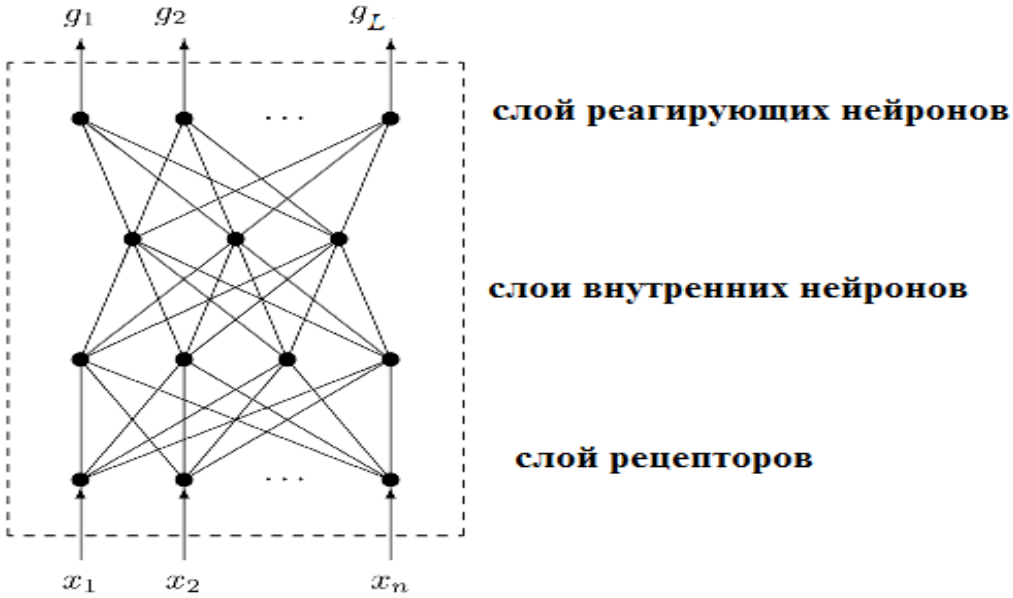
В методе Многослойный перцептрон сеть формируется из нескольких слоёв нейронов. В их число входит слой входных рецепторов, подающих сигналы на нейроны из внутренних слоёв. Слои внутренних нейронов осуществляют преобразование сигналов. Слой реагирующих нейронов производит окончательную классификацию объектов на основании сигналов, поступающих от нейронов, принадлежащих внутренним слоям.
Обычно соблюдаются следующие правила формирования структуры сети.

- Допускаются связи между только между нейронами, находящимися в соседних слоях.
- Связи между нейронами внутри одного слоя отсутствуют.
- Активационные функции для всех внутренних нейронов идентичны.

Для решения задач распознавания с L классами $K_1, \ldots, K_L$ используется конфигурация с L реагирующими нейронами.

Многослойный перцептрон

Схема многослойного перцептрона с двумя внутренними слоями представлена на рисунке 2.



Отметим, что сигналы $g_1, \ldots, g_L$, вычисляемые на выходе реагирующих нейронов, интерпретируются как оценки за классы $K_1, \ldots, K_L$. Весовые коэффициенты w сопоставлены каждой из связей между нейронами из различных слоёв.

Многослойный перцептрон

Рассмотрим процедуру распознавания объектов с использованием многослойного перцептрона. Предположим, что конфигурация нейронной сети включает наряду со слоем рецепторов и слоем реагирующих нейронов также H внутренних слоёв искусственных нейронов. Заданы также количества нейронов в каждом слое. Пусть n – число входных нейронов-рецепторов, $r(h)$ - число нейронов в внутреннем слое h. На первом этапе вектор рецепторы формируют по информации, поступающей из внешней среды, вектор входных переменных (сигналов) $u_1^0, \ldots, u_n^0$. Отметим, что входные сигналы $u_1^1, \ldots, u_n^0$ могут интерпретироваться как признаки $X_1, \ldots, X_n$ в общей постановке задачи распознавания.

Предположим, что для i-го нейрона 1-го внутреннего слоя связь с рецепторами осуществляется с помощью весовых коэффициентов $w_1^{i0}, \ldots, w_n^{i0}$. Сумматор i-го нейрона первого внутреннего слоя вычисляет взвешенную сумму $\xi^{i0} = \sum_{t=0}^{n} w_t^{i0} u_t^0$.

Многослойный перцептрон

Сигнал на выходе i-го нейрона первого внутреннего слоя вычисляется по формуле $u_i^1 = \Phi(\xi^{i0})$. Аналогичным образом вычисляются сигналы на выходе нейронов второго внутреннего слоя. Сигналы $g_1, \ldots, g_L$ рассчитываются с помощью той же самой процедуры, которая используется при вычислении сигналов на выходе нейронов из внутренних слоёв. То есть при вычислении g_i на первом шаге соответствующий сумматор вычисляет взвешенную сумму

$$\xi^{iH} = \sum_{t=0}^{n} w_t^{iH} u_t^H,$$

где $w_1^{iH}, \ldots, w_{r(H)}^{iH}$ - весовые коэффициенты, характеризующие связь i –го реагирующего нейрона с нейронами последнего внутреннего слоя H, $u_1^H, \ldots, u_{r(H)}^H$ - сигналы на выходе внутреннего слоя H. Сигнал на выходе i -го реагирующего нейрона вычисляется по формуле $g_i = \Phi(\xi^{iH})$. Очевидно, что вектор выходных сигналов является функцией вектора входных сигналов $\boldsymbol{u}^0$ (вектора признаков $\boldsymbol{x}$) и матрицы весовых коэффициентов связей между нейронами.

Аппроксимирующие способности многослойных перцептронов

Один реагирующий нейрон позволяет аппроксимировать области, являющиеся полупространствами, ограниченными гиперплоскостями. Нейронная сеть с одним внутренним слоем позволяет аппроксимировать произвольную выпуклую область в многомерном признаковом пространстве (открытую или закрытую). Было доказано также, что МП с двумя внутренними слоями позволяет аппроксимировать произвольные области многомерного признакового пространства. Аппроксимирующая способность способность многослойного перцептрона с различным числом внутренних слоёв проиллюстрирована на рисунке 3. На рисунке области, соответствующие классам ω_1 и ω_2 разделяются с помощью простого нейрона, а также с помощью многослойных перцептронов с одним и двумя внутренними слоями.

Аппроксимирующая способность многослойных перцептронов

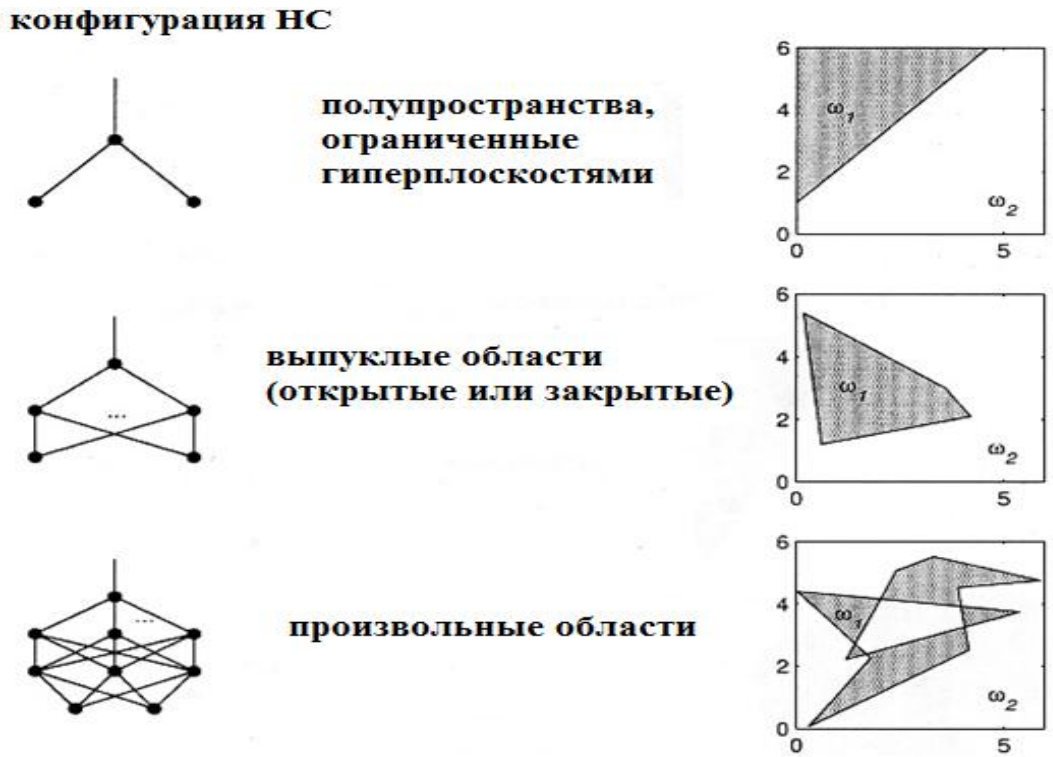


Рис.3

Метод обратного распространения ошибки

Наиболее распространённым способом обучения нейросетевых алгоритмов является метод обратного распространения ошибки. Обозначим через $\boldsymbol{\alpha}_* = (\alpha_{*1}, \ldots, \alpha_{*L})$ вектор индикаторных функций классов $K_1, \ldots, K_L$ на объекте s_* с описанием $\boldsymbol{x}_*$. То есть $\alpha_{*i} = 1$, если $s_* \in K_i$ и $\alpha_{*i} = 0$ в противном случае. Пусть на выходе i-го реагирующего нейрона вычисляется оценка $g_i(\boldsymbol{x}_*)$ за класс K_i , принадлежащая отрезку $[0, 1]$. Отметим, что оценка $g_i(\boldsymbol{x}_*)$ вычисляется активационной функцией реагирующего нейрона. Далее будет предполагаться, что данная активационная функция является сигмоидной. Такое же предположение делается для активационных функций каждого из внутренних нейронов. Потери, связанные с классификацией объекта s_* естественно оценивать с помощью функционала

$$\sum_{i=1}^{L} [\alpha_{*i} - g_i(\boldsymbol{x}_*)]^2.$$

Метод обратного распространения ошибки

Качество аппроксимации на обучающей выборке $\widetilde{S}_t = \{(\boldsymbol{\alpha}_1, \boldsymbol{x}_1), \ldots, (\boldsymbol{\alpha}_m, \boldsymbol{x}_m)\}$ оценивается с помощью функционала

$$E(\widetilde{S}_t, \widetilde{\mathbf{w}}) = \sum_{j=1}^{m} \sum_{i=1}^{L} [\alpha_{ji} - g_i(\boldsymbol{x}_j)]^2.$$

Где $\widetilde{\mathbf{w}} = \{w_t^{ih} \mid h = 0, \ldots, H; t = 1, \ldots, r(h); i = 1, \ldots, (r_{h+1})\}$ - множество весовых коэффициентов связей меду нейронами . Обучение заключается в поиске значений коэффициентов из $\widetilde{\mathbf{w}}$, при которых достигает минимума функционал $E(\widetilde{S}_t, \widetilde{\mathbf{w}})$. В основе обучения лежит метод градиентного спуска. Метод градиентного спуска является итерационным методом оптимизации произвольного функционала F , зависящего от параметров $(\theta_1, \ldots, \theta_r)$ и дифференцируемого по каждому из параметров в произвольной точке $\mathbb{R}^n$. Новые значения вектора параметров на k-ой итерации $\boldsymbol{\theta}^{(k)}$ вычисляется через вектор $\boldsymbol{\theta}^{(k-1)}$, полученный на предыдущей итерации.

При этом используется формула

$$\boldsymbol{\theta}^{(k)} = \boldsymbol{\theta}^{(k-1)} + \eta \times \boldsymbol{grad}[F(\theta_1, \ldots, \theta_r)],$$

где $\eta > 0$ - вещественный параметр, задающий размер каждого шага. На предварительном этапе обучения весовым коэффициентам из $\widetilde{\mathbf{w}}$ случайным образом присваиваются исходные значения. На обучение подаётся некоторый объект обучающей выборки $s_j = (\boldsymbol{\alpha}_j, \boldsymbol{x}_j)$, по описанию которого вычисляются входные и выходные сигналы внутренних нейронов сети, а также выходные сигналы реагирующих нейронов $g_1(\boldsymbol{x}_j, \widetilde{\boldsymbol{w}}), \ldots, g_L(\boldsymbol{x}_j, \widetilde{\boldsymbol{w}})$. Проведём коррекцию весовых коэффициентов связей i-го реагирующего нейрона с нейронами предшествующего внутреннего слоя:

$$(w_0^{iH}, \ldots, w_{r(H)}^{iH}).$$

Для упрощения формул $g_i(\boldsymbol{x}_j, \widetilde{\boldsymbol{w}})$ будем обозначать $g_i(\boldsymbol{x}_j)$ или просто g_i.

Метод обратного распространения ошибки

От весовых коэффициентов $(w_0^{iH}, \ldots, w_{r(H)}^{iH})$ зависит только компонента $[\alpha_{ji} - g_i(\boldsymbol{x}_j)]^2$ ошибки прогнозирования для объекта s_j, равная $E(s_j, \widetilde{\mathbf{w}}) = \sum_{i=1}^{L} [\alpha_{ji} - g_i(\boldsymbol{x}_j)]^2$. Поэтому

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{iH}} = \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial g_i(\boldsymbol{x}_j)} \frac{\partial g_i}{\partial w_t^{iH}} =$$

$$= -2[\alpha_{ji} - g_i(\boldsymbol{x}_j)] \frac{\partial g_i(\boldsymbol{x}_j)}{\partial w_t^{iH}}$$

Однако

$$\frac{\partial g_i(\boldsymbol{x}_j)}{\partial w_t^{iH}} = \frac{\partial g_i(\boldsymbol{x}_j)}{\partial \xi^{iH}} \frac{\partial \xi^{iH}}{\partial w_t^{iH}},$$

где $\xi^{iH} = \sum_{t=1}^{r(H)} w_t^{iH} u_t^H$, u_t^H - сигнал на выходе нейрона с номером t из слоя H.

Метод обратного распространения ошибки

Поскольку g_i является сигмоидной функцией от ξ^{iH}, то

$$\frac{\partial g_i}{\partial w_t^{iH}} = (1 - g_i) g_i u_t^H.$$

Таким образом

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{iH}} = \delta^{iH} u_t^H,$$

где

$$\delta^{iH} = \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial \xi^{iH}} = -2[\alpha_{ji} - g_i(\boldsymbol{x}_j)][1 - g_i(\boldsymbol{x}_j)] g_i(\boldsymbol{x}_j).$$

Воспользовавшись методом градиентного спуска, запишем новые значения весовых коэффициентов $w_t^{iH}(k)$, вычисляемые на k итерации k в виде

$$w_t^{iH}(k) = w_t^{iH}(k-1) + \eta \times \delta^{iH} u_t^H$$

.

Рассмотрим теперь коррекцию весовых коэффициентов $[w_0^{i(H-1)}, \ldots, w_{r(H-1)}^{i(H-1)}]$, соответствующих связям нейрона i из слоя H с нейронами предшествующего внутреннего слоя $(H-1)$. Вклад этих коэффициентов в величину ошибки осуществляется только через сигнал u_i^H на выходе нейрона i из слоя H. Поэтому

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{i(H-1)}} = \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial u_i^H} \frac{\partial u_i^H}{\partial w_t^{i(H-1)}}$$

Однако

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial u_t^H} = \sum_{l=1}^{L} \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial \xi^{lH}} \frac{\partial \xi^{lH}}{\partial u_t^H}$$

Принимая во внимание, что

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial \xi^{lH}} = \delta^{lH},$$

а также, что $\frac{\partial \xi^{lH}}{\partial u_t^H} = w_t^{lH}$,
получаем

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial u_t^H} = \sum_{l=1}^{L} \delta^{lH} w_t^{lH}$$

Исходя из предположения о том, что активационная функция каждого из нейронов является сигмоидной, нетрудно показать также, что

$$\frac{\partial u_i^H}{\partial w_t^{i(H-1)}} = u_i^H (1 - u_i^H) u_t^{H-1}$$

В итоге

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{i(H-1)}} = (\sum_{l=1}^{L} \delta^{lH} w_i^{lH}) u_i^H (1 - u_i^H) u_t^{H-1} = \delta^{i(H-1)} u_t^{H-1},$$

где

$$\delta^{i(H-1)} = \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial \xi^{i(H-1)}} = (\sum_{l=1}^{L} \delta^{lH} w_i^{lH}) u_i^H (1 - u_i^H)$$

Воспользовавшись методом градиентного спуска, запишем новые значения весовых коэффициентов $w_t^{i(H-1)}(k)$, вычисляемые на итерации k в форме

$$w_t^{i(H-1)}(k) = w_t^{i(H-1)}(k-1) + \eta \times \delta^{i(H-1)} u_t^{H-1}$$

Метод обратного распространения ошибки

Рассмотрим теперь процедуру коррекции весовых коэффициентов w для связей между искусственными нейронами из слоя h с искусственными нейронами из слоя $h+1$ при $h < H-1$. Пусть

$$[w_0^{i(H-h)}, \ldots, w_{r(H-h)}^{i(H-h)}]$$

- весовые коэффициенты, связывающие нейрон с номером i из слоя $H-h+1$ с нейронами из слоя $H-h$. Очевидно, что справедливо равенство:

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{i(H-h)}} = \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial u_i^{H-h+1}} \frac{\partial u_i^{H-h+1}}{\partial w_t^{i(H-h)}}$$

Нетрудно показать, что

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial u_i^{H-h+1}} = \sum_{l=1}^{r(H-h+1)} \frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial \xi^{H-h+1}} \frac{\partial \xi^{H-h+1}}{\partial u_i^{H-h+1}} = \sum_{l=1}^{r(H-h+1)} \delta^{l(H-h+1)} w_i^{l(H-h+1)}$$

Учитывая, что активционная функция для кажого внутреннего нейрона является сигмоидной, и принимая во внимание определение $\xi^{i(H-h)}$, получаем

$$\frac{\partial u_i^{H-h+1}}{\partial w_t^{i(H-h)}} = \frac{\partial u_i^{H-h+1}}{\partial \xi^{i(H-h)}} \frac{\partial \xi^{i(H-h)}}{\partial w_t^{i(H-h)}} = u_i^{H-h+1}(1 - u_i^{H-h+1}) u_t^{H-h}$$

В итоге получаем

$$\frac{\partial E(s_j, \widetilde{\mathbf{w}})}{\partial w_t^{i(H-h)}} = \delta^{i(H-h)} u_t^{H-h},$$

где

$$\delta^{i(H-h)} = [\sum_{l=1}^{r(H-h+1)} \delta^{l(H-h+1)} w_i^{l(H-h+1)}] u_i^{H-h+1}(1 - u_i^{H-h+1})$$

Метод обратного распространения ошибки

Коррекция согласно процедуре градиентного спуска производится по формуле:

$$w_t^{i(H-h)}(k) = w_t^{i(H-h)}(k-1) + \eta \times \delta^{i(H-1)} u_t^{H-h}$$

Таким образом может быть представлен общая схема метода обратного распространения ошибки для многослойного перцептрона. На предварительном этапе выбирается архитектура сети: задаётся число внутренних слоёв и количества нейронов в каждом слое. Случайным образом задаются исходные весовые коэффициенты. На вход многослойного перцептрона поочерёдно подаются векторные описания объектов обучающей выборки. С использованием описанного выше способа производится коррекция весовых коэффициентов. С использованием новых скорректированных весовых коэффициентов $\widetilde{\mathbf{w}}$ вычисляется значение функционала $E(s_j, \widetilde{\mathbf{w}})$.

Метод обратного распространения ошибки

Обучение заканчивается при выполнении одного из заранее заданных условий:

- а) Величина функционала ошибки оказывается меньше выбранного порогового значения: $E(s_j, \widetilde{\mathbf{w}}) < \varepsilon$;
- б) Изменения функционала ошибки на протяжений нескольких последних итераций оказывается меньшим некоторого порогового значения.
- в) общее время обучения превышает допустимый предел;

Лекция 7

Ядерные методы, метод опорных векторов

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Ядерные методы

Напомним, что байесовское решающее правило или оптимальное решающее правило в смысле леммы Неймана-Пирсона могут быть легко восстановлены, если для каждого из распознаваемых классов $K_1, \ldots, K_L$ известны соответствующие плотности вероятности $p_1(\boldsymbol{x}), \ldots, p_L(\boldsymbol{x})$. Ранее нами рассматривался методы восстановления плотностей $p_1(\boldsymbol{x}), \ldots, p_L(\boldsymbol{x})$, основанные на гипотезе о независимости переменных $X_1, \ldots, X_n$, а также на гипотезе о нормальности соответствующих распределений. Альтернативным подходом является использование ядерных методов восстановления плотности. Плотность вероятностей для каждого из классов $K_1, \ldots, K_L$ в точках пространства $\mathbb{R}^n$ оценивается по объектам стандартной обучающей выборки $\widetilde{S}_t = \{s_1 = (\boldsymbol{\alpha}_1, \boldsymbol{x}_1), \ldots, s_m = (\boldsymbol{\alpha}_m, \boldsymbol{x}_m)\}$ с использованием ядровых функций.

Плотность вероятности для класса K_i в точке $\boldsymbol{x} \in \mathbb{R}^n$ вычисляется по формуле

$$p_i(\boldsymbol{x}) = \frac{1}{m_i} \sum_{s_j \in K_i} \mathfrak{K}(\boldsymbol{x}, \boldsymbol{x}_j),$$

где m_i - число объектов в обучающей выборки из класса K_i,$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'')$ является неотрицательной вещественнозначной функцией заданной, заданной на декартовом произведении $\mathbb{R}^n \times \mathbb{R}^n$. При этом

- $\mathfrak{K}$ обладает свойством симметричности, то есть

$$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'') = \mathfrak{K}(\boldsymbol{x}'', \boldsymbol{x}');$$

- $\mathfrak{K}$ достигает максимума при $\boldsymbol{x}' = \boldsymbol{x}''$;
- $\mathfrak{K}$ убывает или по крайней мере не возрастает по мере увеличения расстояния между точками $\boldsymbol{x}'$ и $\boldsymbol{x}''$.

Ядерные методы

В одномерном случае, когда оценка плотности вероятности производится в точках интервала допустимых значений одного признака X по обучающей выборке $\{(\boldsymbol{\alpha}_1, x_1), \ldots, (\boldsymbol{\alpha}_m, x_m)\}$, в качестве ядровых используются функции от x вида $\mathbf{K}(\frac{x-x_j}{h})$, где h - параметр сглаживания, характеризующий подробность аппроксимации. Для одномерных ядровых функций предполагается выполнение требований:

- максимальное значение $\mathbf{K}(u)$ должно достигаться в точке 0;
- должно выполняться свойство симметричности $\mathbf{K}(u) = \mathbf{K}(-u)$;
- $\mathbf{K}(u)$ монотонно убывающая или не возрастающая по мере удаления u от 0 функция;
- требуется выполнение равенства $\int_{-\infty}^{\infty} \mathbf{K}(u)du = 1$.

Очевидно, что такими свойствами обладают:

- прямоугольное ядро $\mathbf{K}(u) = \frac{1}{2}$ при $\mid u \mid \leq 1$ и $\mathbf{K}(u) = 0$ при $\mid u \mid > 1$;
- гауссиана $\mathbf{K}(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}$.

Перечисленными свойствами обладает также ядро Епанечникова $\mathbf{K}(u) = \frac{3}{4}(1-u^2)$ при $\mid u \mid \leq 1$ и $\mathbf{K}(u) = 0$ при $\mid u \mid > 1$. Параметр масштаба h характеризует подробность аппроксимации. При $h \to 0$ ядерный аппроксиматор подробно описывает детали эмпирического распределения объектов обучающей выборки. Однако устойчивость аппроксимации при этом теряется. При увеличении h возрастает устойчивость аппроксимации, но теряется способность улавливать детали.

Многомерные ядра размерности n могут задаваться в виде произведения одномерных ядер:

$$\mathfrak{K}(\boldsymbol{x}_*, \boldsymbol{x}_j) = \frac{1}{h^n} \prod_{i=1}^{n} \mathbf{K}\left(\frac{x_{*i} - x_{ji}}{h}\right).$$

Ядерные методы

Используются также ядровые функции, где связь между точками задаётся через расстояние между ними. Например, могут использоваться ядровые функции типа гауссианы

$$\mathfrak{K}(\boldsymbol{x}_*, \boldsymbol{x}_j) = \sqrt{2\pi}\sigma^n e^{-\frac{(\boldsymbol{x}_* - \boldsymbol{x}_j)^2}{2\sigma^2}}.$$

Согласно формуле Байеса распознаваемый объект относится в класс, для которого величина $p_i(\boldsymbol{x})P(K_i)$ максимальна . Параметр сглаживания h может подбираться таким образом, чтобы точность распознавания была максимальной. Например, может максимизироваться оценка точности на контрольной выборке или в оценкой скользящего контроля.

Метод опорных векторов является универсальным методом распознавания, позволяющим наряду с линейными реализовывать также нелинейные решающие правила. Исходный вариант метода был предложен для задач с двумя распознаваемыми классами K_1 и K_2. В случаях, когда объекты разных классов в обучающей выборке линейно разделимы, обычно существует целая совокупность линейных поверхностей, осуществляющих такое разделение. На рисунке представлены двумерные данные, где объекты двух классов могут быть раделены с помощью прямых A, B, C, D. Однако наша интуиция, подсказывает что наилучшей обобщающей способностью должна обладать разделяющая прямая F, одинаково удалённая от групп объектов из разных классов.

Метод опорных векторов

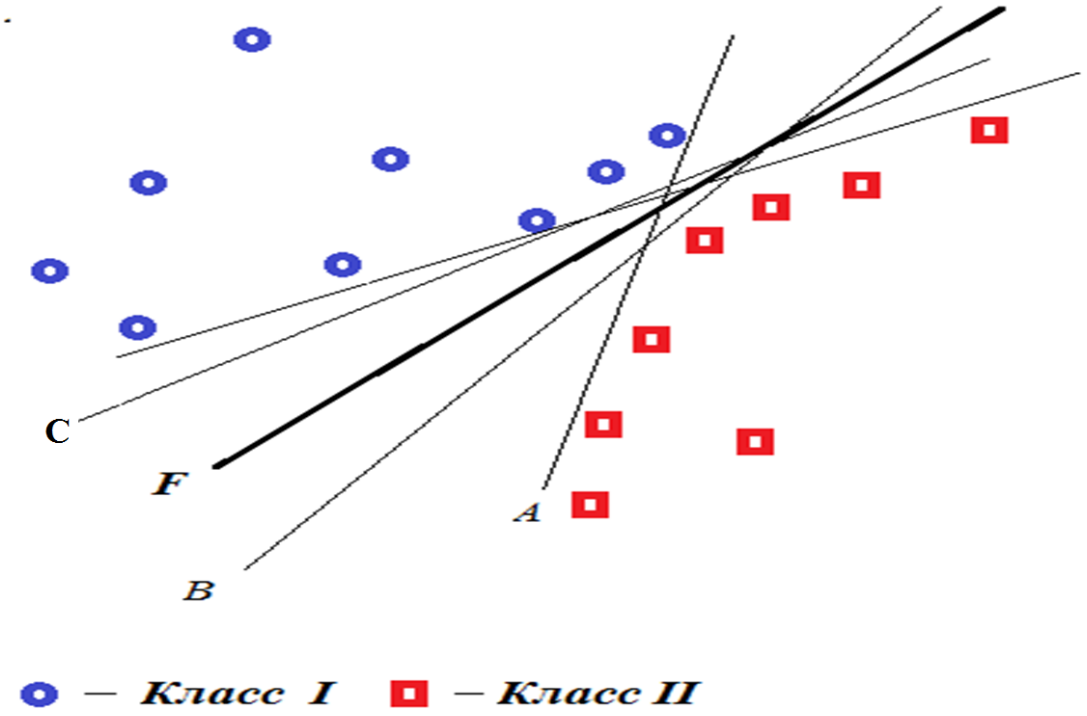


Рис.1

Интуитивные представления об оптимальной разделимости формализует проведение разделяющей гиперплоскости посередине между двумя параллельными гиперплоскостями, каждая из которых отделяет объекты одного из классов.

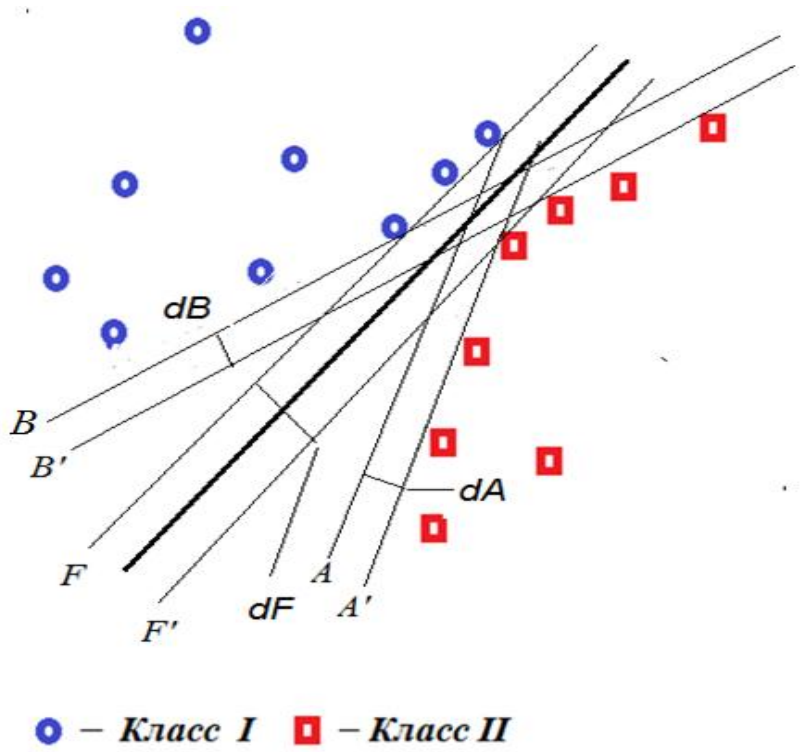


Рис.2

При этом две плоскости строятся таким образом, чтобы расстояние «зазор» между ними был бы максимальным. Из рисунка видно, что наибольшим является «зазор» между двумя параллеь ными прямыми F и F'.

Напомним, что пара параллельных гиперплоскостей P_1 и P_2 в многомерном пространстве $\mathbb{R}^n$ описывается с помощью уравнений

$$P_1 \to \boldsymbol{w}\boldsymbol{x}^t = b_1 \tag{1}$$

$$P_2 \to \boldsymbol{w}\boldsymbol{x}^t = b_2$$

От системы (1) нетрудно перейти к эквивалентной системе

$$\boldsymbol{z}\boldsymbol{x}^t = b + 1 \tag{2}$$

$$\boldsymbol{z}\boldsymbol{x}^t = b - 1,$$

описывающей те же самые гиперплоскости. Расстояние (величина зазора) δ между гиперплоскостями P_1 и P_2 равно $\frac{2}{|\boldsymbol{z}|}$.

Следовательно задача поиска двух максимально удалённых друг от друга параллельных гиперплоскостей, каждая из которых отделяет объекты одного из классов, может быть сведена к оптимизационной задаче с ограничениями

$$\delta = \frac{2}{|\,\boldsymbol{z}\,|} \to \max \tag{3}$$

$$\boldsymbol{z}\boldsymbol{x}_j^t \geq b + 1 \text{ при } s_j \in K_1 \bigcap \widetilde{S}_t,$$
$$\boldsymbol{z}\boldsymbol{x}_j^t \leq b - 1 \text{ при } s_j \in K_2 \bigcap \widetilde{S}_t.$$

При этом оптимизация производится по компонентам направляющего вектора $\boldsymbol{z} = (z_1, \ldots, z_n)$ и параметру сдвига b. Введём обозначение

$$\alpha_j = 1 \text{ при } s_j \in K_1 \text{ и}$$
$$\alpha_j = -1 \text{ при } s_j \in K_2$$

Тогда задача (3) оказывается эквивалентна задаче

$$\frac{1}{2}\sum_{i=1}^{n} z_i^2 \to \min \tag{4}$$

$$\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) \geq 1, j = 1, \ldots, m$$

Из известной теоремы Каруша-Куна-Такера (KKT) следует, что для произвольной точки $(\boldsymbol{z}^*, b^*)$, в которой $\sum_{i=1}^{n} z_i^2$ достигает своего минимума при ограничениях задачи (4), и некоторого вектора неотрицательных множителей Лагранжа $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)$ соблюдаются условия стационарности лагранжиана

$$L(\boldsymbol{z}, b, \boldsymbol{\lambda}) = \frac{1}{2}\sum_{i=1}^{n} z_i^2 - \sum_{j=1}^{m} \lambda_j[\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1]$$

Также из теоремы KKT следует необходимость выполнения m равенств, которые носят название условий дополняющей нежёсткости

$$\lambda_j[\alpha_j(\boldsymbol{z}^*\boldsymbol{x}_j^t - b^*) - 1] = 0, j = 1, \ldots, m$$

Из условия стационарности следует, что

$$\frac{\partial L(\boldsymbol{z}, b, \boldsymbol{\lambda})}{\partial z_i}|_(\boldsymbol{z}^*, b^*) = z_i^* - \sum_{j=1}^{m} \lambda_j \alpha_j x_{ji} = 0 \tag{5}$$

В векторной форме система (5) принимает вид

$$\boldsymbol{z}^* = \sum_{j=1}^{m} \lambda_j \alpha_j \boldsymbol{x}_j$$

,

Из условия стационарности следует выполнение равенства

$$\frac{\partial L(\boldsymbol{z}, b, \boldsymbol{\lambda})}{\partial b} = \sum_{j=1}^{m} \lambda_j \alpha_j = 0 \tag{6}$$

Таким образом для получения решения задачи (4) достаточно найти требуемый вектор значений неотрицательных множителей Лагранжа $\boldsymbol{\lambda}^* = (\lambda_1^*, \ldots, \lambda_m^*)$.

Поиск оптимальных значений множителей Лагранжа.

Нетрудно показать, что лагранжиан в произвольной точке $(\boldsymbol{z}, b, \boldsymbol{\lambda})$, для которой соблюдаются условия (5,6), может быть записан в виде

$$L(\boldsymbol{z}, b, \boldsymbol{\lambda}) = g(\boldsymbol{\lambda}) = \sum_{j=1}^{m} \lambda_j - \frac{1}{2} \sum_{j'=1}^{m} \sum_{j''=1}^{m} \lambda_{j'} \lambda_{j''} \alpha_{j'} \alpha_{j''} (\boldsymbol{x}_{j'} \boldsymbol{x}_{j''}^t).$$

Отметим, что в силу соблюдения ограничений задачи (4) и неотрицательности множителей Лагранжа выполняется неравенство

$$g(\boldsymbol{\lambda}) \leq \sum_{i=1}^{n} z_i^2.$$

Из теории оптимизации с ограничениями следует, что при соблюдении ряда условий, которые справедливы для задачи (4), максимум $g(\boldsymbol{\lambda})$ по вектору множителей Лагранжа при условии их неотрицательности и при соблюдении равенства $\sum_{j=1}^{m} \lambda_j \alpha_j = 0$ равен $\frac{1}{2} \sum_{i=1}^{n} (z_i^*)^2$. Однако в силу соблюдения условий стационарности и условий дополняющей нежёсткости справедливо равенство

$$L(\boldsymbol{z}^*, b^*, \boldsymbol{\lambda}) = g(\boldsymbol{\lambda}^*) = \frac{1}{2} \sum_{i=1}^{n} (z_i^*)^2.$$

То есть точка $\boldsymbol{\lambda}^*$ является точкой максимума $g(\boldsymbol{\lambda})$ при условии неотрицательности $(\lambda_1, \ldots, \lambda_m)$ и при соблюдении равенства $\sum_{j=1}^{m} \lambda_j \alpha_j = 0$.

Таким образом оказывается, что оптимальные значения множителей множителей Лагранжа $(\lambda_1, \ldots, \lambda_m)$ могут быть найдены как решение двойственной задачи квадратичного программирования:

$$\sum_{j=1}^{m} \lambda_j - \frac{1}{2} \sum_{j'=1}^{m} \sum_{j''=1}^{m} \lambda_{j'} \lambda_{j''} \alpha_{j'} \alpha_{j''} (\boldsymbol{x}_{j'} \boldsymbol{x}_{j''}^t) \to \max \tag{7}$$

$$\sum_{j=1}^{m} \lambda_j \alpha_j = 0$$

$$\lambda_j \geq 0, j = 1, \ldots, m$$

Метод опорных векторов

Пусть $(\hat{\lambda}_1, \ldots, \hat{\lambda}_m)$ - решение задачи (7) . Направляющий вектор оптимальной разделяющей гиперплоскости находится по формуле $\sum_{j=1}^{m} \hat{\lambda}_j \alpha_j \boldsymbol{x}_j$ То есть направляющий вектор разделяющей гиперплоскости является линейной комбинацией векторных описаний объектов обучающей выборки, для которых значения соответствующих оптимальных множителей Лагранжа отличны от 0. Такие векторные описания принято называть опорными векторами. Пусть

$$J_0 = \{j = 1, \ldots, m \mid [\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1] \neq 0\}$$

Из условий дополняющей нежёсткости видно, при $j \in J_0$ обязательно должно выполняться равенство $\hat{\lambda}_j = 0$. Поэтому векторное описание $\boldsymbol{x}_j$ соответствующего объекта обучающей выборки является опорным вектором, если j не принадлежит J_0. Оценка параметра сдвига $\hat{b}$ находится из ограничения, соответствующего произвольному опорному вектору.

Метод опорных векторов

Действительно, из условий дополняющей нежёсткости следует, что при произвольном j, когда $\lambda_j > 0$, обязательно должно выполняться равенство $\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1 = 0$, эквивалентное равенству $b = \boldsymbol{z}\boldsymbol{x}_j^t - \alpha_j$. Классификация нового распознаваемого объекта s с описанием $\boldsymbol{x}$ вычисляется согласно знаку выражения

$$g(\boldsymbol{x}) = \sum_{j=1}^{m} \hat{\lambda}_j \alpha_j (\boldsymbol{x}_j \boldsymbol{x}^t) - \hat{b}. \tag{8}$$

Объект s относится к классу K_1, если $g(\boldsymbol{x})$ и объект s относится к классу K_2 в противном случае.
Существенным недостатком рассмотренного варианта метода опорных векторов является требование линейной разделимости классов.
Однако данный недостаток может быть легко преодолён с помощью следующей модификации, основанной на использовании дополнительного вектора неотрицательных переменных $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_m)$.

Метод опорных векторов. Случай отсутствия линейной разделимости.

Требования об отделимости классов (3) заменяются более мягкими требованиями:

$$\boldsymbol{z}\boldsymbol{x}_j^t \geq b + 1 - \xi_j \text{ при } s_j \in K_1 \bigcap \widetilde{S}_t,$$
$$\boldsymbol{z}\boldsymbol{x}_j^t \leq b - 1 + \xi_j \text{ при } s_j \in K_2 \bigcap \widetilde{S}_t.$$

Выдвигается также требование минимальности суммы $\sum_{j=1}^{m} \xi_j$. Поиск оптимальных параметров разделяющей гиперплоскости при отсутствии линейной разделимости таким образом сводится к решению задачи квадратично программирования

$$\frac{1}{2}\sum_{i=1}^{n} z_i^2 + C\sum_{j=1}^{m} \xi_j \to \min \tag{9}$$

$$\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) \geq 1 - \xi_j, \xi_j \geq 0, j = 1, \ldots, m$$

где - некоторая положительная константа, являющаяся открытым параметром алгоритма. Иными словами оптимальное значение C подбирается пользователем.

Из теоремы KKT следует, что для произвольной точки $(\boldsymbol{z}^*, b^*, \boldsymbol{\xi}^*)$, в которой достигается минимум функционала

$$\frac{1}{2}\sum_{i=1}^{n} z_i^2 + C\sum_{j=1}^{m}\xi_j$$

при справедливости ограничений (9) , и некоторых векторов неотрицательных множителей Лагранжа $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)$ и $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_m)$ соблюдаются условия стационарности лагранжиана

$$L(\boldsymbol{z}, b, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{1}{2}\sum_{i=1}^{n} z_i^2 - \sum_{j=1}^{m}\lambda_j[\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1 + \xi_j] - \sum_{j=1}^{m}\eta_j\xi_j$$

Метод опорных векторов. Случай отсутствия линейной разделимости.

Данные условия записываются в виде

$$\frac{\partial L(\boldsymbol{z}, b, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta})}{\partial z_i} \Big|_{z_i^*} = z_i^* - \sum_{j=1}^{m} \lambda_j \alpha_j x_{ji} = 0 \tag{10}$$

$$i = 1, \ldots, n$$

$$\frac{\partial L(\boldsymbol{z}, b, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta})}{\partial b} = \sum_{j=1}^{m} \lambda_j \alpha_j = 0,$$

$$\frac{\partial L(\boldsymbol{z}, b, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta})}{\partial \xi_j} = C - \lambda_j - \eta_j = 0,$$

$$j = 1, \ldots, m.$$

Метод опорных векторов. Случай отсутствия линейной разделимости.

Также выполняются условия дополняющей нежёсткости

$$\lambda_j[\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1 + \xi_j] = 0, \tag{11}$$

$$\eta_j\xi_j = 0,$$

$$j = 1, \ldots, m$$

Оптимальные значения множителей $(\lambda_1, \ldots, \lambda_m)$ могут быть найдены как решение двойственной задачи квадратичного программирования

$$\sum_{j=1}^{m}\lambda_j - \frac{1}{2}\sum_{j'=1}^{m}\sum_{j''=1}^{m}\lambda_{j'}\lambda_{j''}\alpha_{j'}\alpha_{j''}(\boldsymbol{x}_{j'}\boldsymbol{x}_{j''}^t) \to \max \tag{12}$$

$$\sum_{j=1}^{m}\lambda_j\alpha_j = 0$$

$$C \geq \lambda_j \geq 0, j = 1, \ldots, m$$

Метод опорных векторов. Случай отсутствия линейной разделимости.

Как и в случае линейной разделимости направляющий вектор оптимальной разделяющей гиперплоскости находится по формуле $\hat{\boldsymbol{z}} = \sum_{i=1}^{m} \hat{\lambda}_j \alpha_j \boldsymbol{x}_j$. Из условий $C - \lambda_j - \eta_j$ и $\eta_j \xi_j = 0$ следует что $\eta_j > 0$ и $\xi_j = 0$ при $0 < \lambda_j < C$. Также как и в случае существования линейной разделимости параметра сдвига $\hat{b}$ находится из ограничения, соответствующего произвольному опорному вектору. Действительно, из условий дополняющей нежёсткости и и следующего из них равенства $\xi_j = 0$ следует выполнение равенства $\alpha_j(\boldsymbol{z}\boldsymbol{x}_j^t - b) - 1 = 0$, эквивалентного равенству $b = \boldsymbol{z}\boldsymbol{x}_j^t - \alpha_j$. Распознавание нового объекта s производится по его описанию $\boldsymbol{x}$ также как и в случае линейно разделимых классов с помощью решающего правила (8) по величине распознающей функции $g(\boldsymbol{x})$.

Метод опорных векторов. Построение нелинейных разделяющих поверхностей

Таким образом построение оптимального решающего правила сводится к решению двойственных задач квадратичного программирования (7) или (12). Следует отметить, что вектора описаний объектов обучающей выборки $\{\boldsymbol{x}_j \mid j = 1, \ldots, m\}$ входят в задачи (7),(12) только через свои скалярные произведения $\boldsymbol{x}_{j'}\boldsymbol{x}_{j''}^t$ при $\lambda_{j'} > 0$ и $\lambda_{j''} > 0$. Аналогично при вычислении значения распознающей функции (8) по описанию распознаваемого объекта $\boldsymbol{x}$ на самом деле используются только скалярные произведения $\boldsymbol{x}\boldsymbol{x}_{j''}^t$. Предположим что в исходном признаковом пространстве эффективное линейное разделение отсутствует. Однако может существовать такое евклидово пространство H_y и такое отображение Φ из области пространства $\mathbb{R}^n$, содержащей описания распознаваемых объектов, в пространство H_y, что образы объектов обучающей выборки из классов K_1 и K_2 оказываются разделимыми с помощью некоторой гиперплоскости P_y. Пусть $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m\}$ -образы в пространстве H_y векторов описаний объектов обучающей выборки $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$.

Метод опорных векторов. Построение нелинейных разделяющих поверхностей

Линейная разделимость означает существование решения аналога задачи квадратичного программирования (4) для пространства H_y, которое сводится к решению двойственной задачи

$$\sum_{j=1}^{m} \lambda_j - \frac{1}{2} \sum_{j'=1}^{m} \sum_{j''=1}^{m} \lambda_{j'} \lambda_{j''} \alpha_{j'} \alpha_{j''} (\boldsymbol{y}_{j'} \boldsymbol{y}_{j''}^t) \to \max \tag{13}$$

$$\sum_{j=1}^{m} \lambda_j \alpha_j = 0$$

$$\lambda_j \geq 0, j = 1, \ldots, m$$

Отметим, что необходимость полного восстановления преобразования $\Phi(\boldsymbol{x})$ для поиска всех коэффициентов задачи квадратичного программирования (13) отсутствует. Достаточно найти функцию, связывающую скалярное произведение $\boldsymbol{y}_{j'} \boldsymbol{y}_{j''}^t$ с векторами $\boldsymbol{x}_{j'}$ и $\boldsymbol{x}_{j''}$, где $\boldsymbol{y}_{j'} = \Phi(\boldsymbol{x}_{j'})$ и $\boldsymbol{y}_{j'} = \Phi(\boldsymbol{x}_{j''})$.

Такую функцию мы далее будем называть потенциальной и обозначать $\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'')$. Можно подобрать потенциальную функцию таким образом, чтобы решение (13) было оптимальным. При этом поиск оптимальной потенциальной функции может производится внутри некоторого заранее заданного семейства. Например, потенциальную функцию можно задать с помощью простого сдвига

$$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'') = \boldsymbol{x}'(\boldsymbol{x}'')^t + \theta, \tag{14}$$

Решение, полученное путём замены скалярных произведений на потенциальные функции, может рассматриваться как построении линейной разделяющей поверхности в трансформированном пространстве, если удаётся доказать существование отображения Φ, для которого при произвольных $\boldsymbol{x}'$ и $\boldsymbol{x}''$ из $\mathbb{R}^n$ выполняется равенство

$$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'') = \Phi(\boldsymbol{x}')\Phi^t(\boldsymbol{x}''). \tag{15}$$

Метод опорных векторов. Построение нелинейных разделяющих поверхностей

Существование преобразования Φ , для которого выполняется равенство (15), было показано для неотрицательных симметричных потенциальных функций вида

$$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'') = (\boldsymbol{x}'(\boldsymbol{x}'')^t + \theta)^2,$$

,где $d \geq 1$-целое число а также ядровых функции типа гауссианы

$$\mathfrak{K}(\boldsymbol{x}', \boldsymbol{x}'') = \sqrt{2\pi}\sigma^n e^{-\frac{(\boldsymbol{x}' - \boldsymbol{x}'')^2}{2\sigma^2}},$$

где σ - вещественная неотрицательная константа (размер ядра). Поскольку в общем случае преобразование является нелинейным, то прообразом в пространстве $\mathbb{R}^n$ линейной разделяющей гиперплоскости, существующей в пространстве H_y , может оказаться нелинейная поверхность.

Метод опорных векторо. Построение нелинейных разделяющих поверхностей

Для большого числа прикладных задач линейная разделимость является недостижимой. Поэтому выбор ядровой функции может производиться из требования о минимальности числа ошибок в смысле задачи квадратичного програмирования (9). На практике подбор ядровых функций и их параметров производится исходя из требования достижения максимальной обобщающей способности, которая оценивается с помощью скользящего контроля или оценок на контрольной выборке.

Метод опорных векторов. Регрессия.

Методика улучшения обобщающей способности, лежащая в основе Метода опорных векторов (МОВ) может быть распространена также на задачи регрессии, то есть на задачи прогнозирования некоторой переменной Y , принимающей значения из интервала вещественной оси по значениями вещественных переменных $X_1, \ldots, X_n$. Вместо требования максимизации величины «зазора» между распознаваемыми классами для задач распознавания в случае задач регрессии выдвигается требование минимизации вариации прогнозирующей функции на области $\widetilde{X}$, из которой принимают значения переменные $X_1, \ldots, X_n$ Уменьшение вариации прогнозирующей функции очевидно позволяет снизить вариационную составляющую обобщённой ошибки прогнозирования и уменьшить эффект переобучения. Предположим, что прогнозом величины Y являются значения функции $f(\boldsymbol{x})$.

Задача снижения вариации функции f, формализуется как задача максимизации параметра

$$\delta_\varepsilon = \inf_{(\boldsymbol{x}', \boldsymbol{x}'') \in \widetilde{X}_\varepsilon^C} (\mid \boldsymbol{x}' - \boldsymbol{x}'' \mid) \tag{16}$$

где ε - некоторый пороговый параметр;

$$\widetilde{X}_\varepsilon^C = \{(\boldsymbol{x}', \boldsymbol{x}'') \in \widetilde{X} \times \widetilde{X} \mid\mid f(\boldsymbol{x}') - f(\boldsymbol{x}'') \mid > 2\varepsilon\}$$

Предположим, что регрессия является линейной, то есть $f(\boldsymbol{x}) = \boldsymbol{\beta}\boldsymbol{x}^t + \beta_0$. , где $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_n)$ - вектор регрессионых коэффициентов, β_0 - параметр сдвига. Откуда следует, что

$$\mid f(\boldsymbol{x}') - f(\boldsymbol{x}'') \mid = \mid \boldsymbol{\beta}(\boldsymbol{x}' - \boldsymbol{x}'')^t \mid$$

$\rightarrow$

Очевидно, что минимум $| \boldsymbol{x}' - \boldsymbol{x}'' |$ достигается для пары векторов из $\widetilde{X}_\varepsilon^C$, для которой

- справедливо равенство $| \boldsymbol{\beta}(\boldsymbol{x}' - \boldsymbol{x}'')^t | = 2\varepsilon$;
- вектор $(\boldsymbol{x}' - \boldsymbol{x}'')$ совпадает по направлению с вектором $\boldsymbol{\beta}$.

В результате должно выполняться равенство

$$| \boldsymbol{\beta} | \, \delta_\varepsilon = 2\varepsilon,$$

эквивалентное равенству

$$\delta_\varepsilon = \frac{2\varepsilon}{| \boldsymbol{\beta} |}.$$

Наряду с требованиями максимизации параметра δ_ε выдвигается также требование точности аппроксимации на обучающей выборке.

Отклонение прогнозирующей функции $f(\boldsymbol{x})$ от значений прогнозируемой величины Y не должно превышать порогового параметра ε . Отметим, что задача максимизации $\delta_\varepsilon = \frac{2\varepsilon}{|\boldsymbol{\beta}|}$ полностью эквивалентна задаче минимизации $\frac{1}{2}\sum_{i=1}^{n}\beta_i^2$.
В результате мы переходим к задаче квадратичного программирования

$$\frac{1}{2}\sum_{i=1}^{n}\beta_i^2 \to \min \tag{17}$$

$$y_j - \beta_0 - \boldsymbol{\beta}\boldsymbol{x}_j^t \le \varepsilon$$

$$\beta_0 + \boldsymbol{\beta}\boldsymbol{x}_j^t - y_j \le \varepsilon, 1, \ldots, m$$

Решение задачи (17) может отсутствовать, если не будет найден вектор $\boldsymbol{\beta}$, при котором справедливы ограничения (17).

Поэтому от задачи (17) переходим к задаче, допускающей существование решений в произвольном случае:

$$\frac{1}{2}\sum_{i=1}^{n}\beta_i^2 + C\sum_{j=1}^{m}(\xi_j^1 + \xi_j^2) \to \min \tag{18}$$

$$y_j - \beta_0 - \boldsymbol{\beta}\boldsymbol{x}_j^t \le \varepsilon + \xi_j^1$$

$$\beta_0 + \boldsymbol{\beta}\boldsymbol{x}_j^t - y_j \le \varepsilon + \xi_j^2, j = 1, \ldots, m$$

где $(\xi_j^1, \xi_j^2), j = 1, \ldots, m$ - неотрицательные коэффициенты, имеющие тот же самый смысл, что и аналогичные коэффициенты в задаче (9). Параметр C - неотрицательный штрафной коэффициент.
Для решения задачи квадратичного программирования (18) используются методы, аналогичные тем, которые используются для решения задачи квадратичного программирования (9), лежащей в основе процедуры обучения алгоритмов распознавания.

Метод опорных векторо. Регрессия.

Подобно тому как вариант МОВ для решения задач распознавания допускает расширение на случаи с линейно неотделимыми классами и принципиально позволяет строить нелинейные разделяющие поверхности, вариант МОВ для решения задач регрессионного анализа допускает расширение на задачи, в которых присутствуют выпадающие наблюдения, а также позволяет строить нелинейные прогнозирующие функции.

Лекция 8

Решающие деревья

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Решающие деревья

Решающие деревья воспроизводят логические схемы, позволяющие получить окончательное решение о классификации объекта с помощью ответов на иерархически организованную систему вопросов. Причём вопрос, задаваемый на последующем иерархическом уровне, зависит от ответа, полученного на предыдущем уровне. Подобные логические модели издавна используются в ботанике, зоологии, минералогии, медицине и других областях. Пример, решающего дерева, позволяющая грубо оценить стоимость квадратного метра жилья в предполагаемом городе приведена на рисунке 1. Схеме принятия решений, изображённой на рисунке 1, соответствует связный ориентированный ациклический граф – ориентированное дерево. Дерево включает в себя корневую вершину, инцидентную только выходящим рёбрами, внутренние вершины, инцидентную одному входящему ребру и нескольким выходящим, и листья – концевые вершины, инцидентные только одному входящему ребру.

.

Решающие деревья

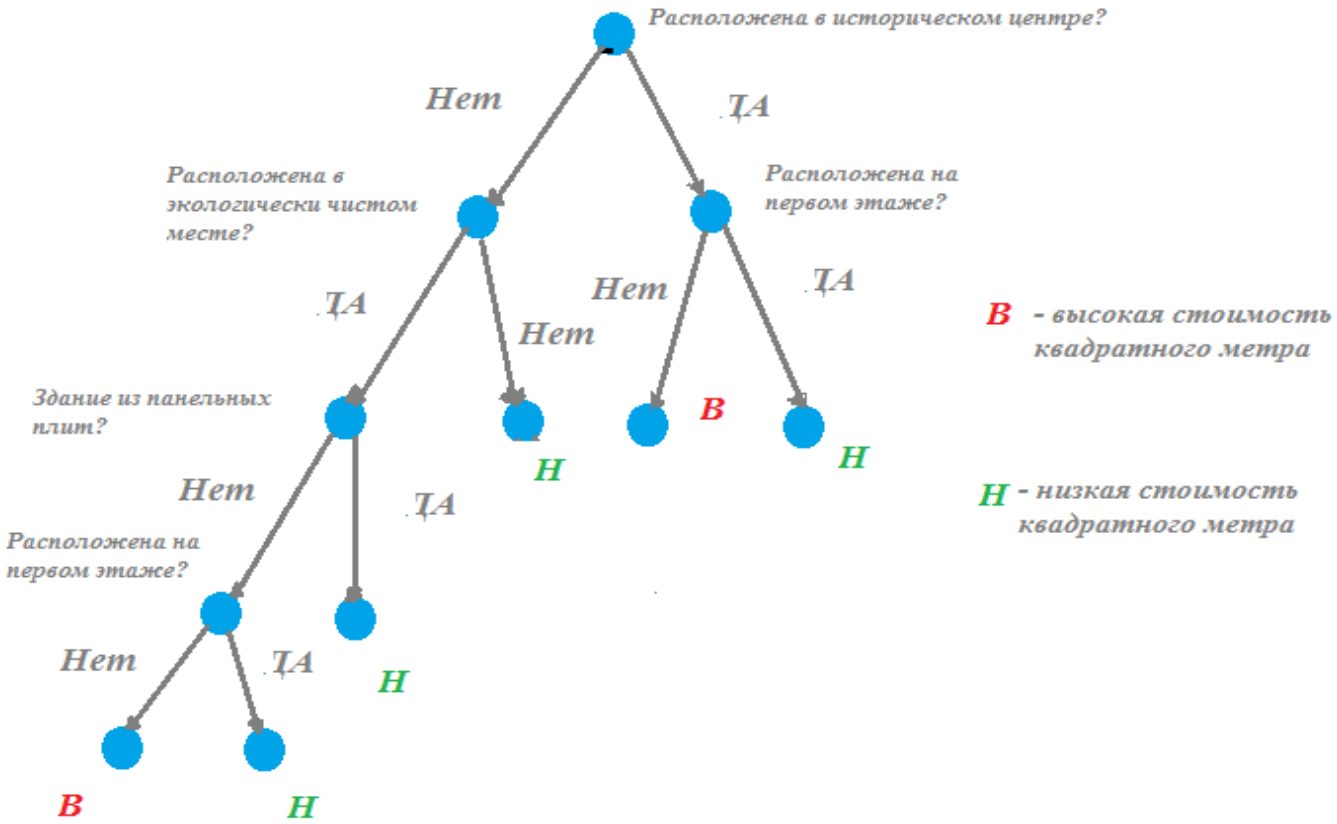


Рис.1

Решающие деревья

Каждой из вершин дерева за исключением листьев соответствует некоторый вопрос, подразумевающий несколько вариантов ответов, соответствующих выходящим рёбрам. В зависимости от выбранного варианта ответа осуществляется переход к вершине следующего уровня. Концевым вершинам поставлены в соответствие метки, указывающие на отнесение распознаваемого объекта к одному из классов. Решающее дерево называется бинарным, если каждая внутренняя или корневая вершина инцидентна только двум выходящим рёбрам. Бинарные деревья удобно использовать в моделях машинного обучения.

Распознавание с помощью решающих деревьев. Предположим, что бинарное дерево T используется для распознавания объектов, описываемых набором признаков $X_1, \ldots, X_n$.

Каждой вершине ν дерева T ставится в соответствие предикат, касающийся значения одного из признаков. Непрерывному признаку X_j соответствует предикат вида "$X_j \geq \delta_j^\nu$", где δ_j^ν - некоторый пороговый параметр.

Решающие деревья. Обучение

Категориальному признаку $X_{j'}$, принимающему значения из множества $M_{j'} = \{a_1^{j'}, \ldots, a_{r(j')}^{j'}\}$ ставится в соответствие предикат вида "$X_{j'} \in M_{j'}^{\nu 1}$" , где $M_{j'}^{\nu 1}$ является элементом дихотомического разбиения $\{M_{j'}^{\nu 1}, M_{j'}^{\nu 2}\}$ множества $M_{j'}$. Выбор одного из двух, выходящих из вершины ν рёбер производится в зависимости от значения предиката.

Процесс распознавания заканчивается при достижении концевой вершины (листа). Объект относится классу согласно метке, поставленной в соответсттвие данному листу.

Обучение решающих деревьев Рассмотрим задачу распознавания с классами $K_1, \ldots, K_L$. Обучение производится по обучающей выборке $\widetilde{S}_t$ и включает в себя поиск оптимальных пороговых параметров или оптимальных дихотомических разбиений для признаков $X_1, \ldots, X_n$. При этом поиск производится исходя из требования снижения среднего индекса неоднородности в выборках, порождаемых искомым дихотомическим разбиением обучающей выборки $\widetilde{S}_t$.

Индекс неоднородности вычисляется для произвольной выборки $\widetilde{S}$, содержащей объекты из классов $K_1, \ldots, K_L$. При этом используется несколько видов индексов, включая:

- энтропийный индекс неоднородности,
- индекс Джини,
- индекс ошибочной классификации.

Энтропийный индекс неоднородности вычисляется по формуле

$$\gamma_e(\widetilde{S}) = -\sum_{i=1}^{L} P_i \ln P_i, \tag{1}$$

где P_i - доля объектов класса K_i в выборке $\widetilde{S}$. При этом принимается, что $0 \ln(0) = 0$. Наибольшее значение $\gamma_e(\widetilde{S})$ принимает при равенстве долей классов. Наименьшее значение $\gamma_e(\widetilde{S})$ достигается при принадлежности всех объектов одному классу.

Решающие деревья. Обучение

Индекс Джини вычисляется по формуле

$$\gamma_g(\widetilde{S}) = 1 - \sum_{i=1}^{L} P_i^2. \tag{2}$$

Индекс ошибочной классификации вычисляется по формуле

$$\gamma_m(\widetilde{S}) = 1 - \max_{1,\ldots,L}(P_i). \tag{3}$$

Нетрудно понять, что индексы (2) и (3) также достигают минимального значения при принадлежности всех объектов обучающей выборке одному классу.
Предположим, что в методе обучения используется индекс неоднородности γ_*. Для оценки эффективности разбиения обучающей выборки $\widetilde{S}_t$ на непересекающиеся подвыборки $\widetilde{S}_t^l$ и $\widetilde{S}_t^r$ используется уменьшение среднего индекса неоднородности в $\widetilde{S}_t^l$ и $\widetilde{S}_t^r$ по отношению к $\widetilde{S}_t$

Решающие деревья. Обучение

Данное уменьшение вычисляется по формуле

$$\Delta(\gamma_*, \widetilde{S}_t) = \gamma_*(\widetilde{S}_t) - P_l\gamma_*(\widetilde{S}_t^l) - P_r\gamma_*(\widetilde{S}_t^r),$$

где P_l и P_r являются долями $\widetilde{S}_t^l$ и $\widetilde{S}_t^r$ в выборке $\widetilde{S}_t$. На первом этапе обучения бинарного решающего дерева ищется оптимальный предикат соответствующий корневой вершине. С этой целью оптимальные разбиения строятся для каждого из признаков из набора $X_1, \ldots, X_n$. Выбирается признак $X_{i_{max}}$ с максимальным значением индекса $\Delta(\gamma_*, \widetilde{S}_t)$. Подвыбороки $\widetilde{S}_t^l$ и $\widetilde{S}_t^r$, задаваемые оптимальным предикатом для $X_{i_{max}}$ оцениваются с помощью критерия остановки. В качестве критерия остановки может быть использован простейший критерий достижения полной однородности по одному из классов. В случае, если какая-нибудь из выборок $\widetilde{S}_t^*$ удовлетворяет критерию остановки, то соответствующая вершина дерева объявляется концевой и для неё вычисляется метка класса. В случае, если выборка $\widetilde{S}_t^*$ не удовлетворяет критерию остановки, то формируется новая внутренняя вершина, для которой процесс построения дерева продолжается.

Решающие деревья. Обучение

Однако вместо обучающей выборки $\widetilde{S}_t$ используется соответствующая вновь образованной внутренней вершине ν выборка $\widetilde{S}_\nu$, которая равна $\widetilde{S}_t^*$. Для данной выборки производятся те же самые построения, которые на начальном этапе проводились для обучающей выборки $\widetilde{S}_t$. Обучение может проводиться до тех пор, пока все вновь построенные вершины не окажутся однородными по классам. Такое дерево может быть построено всегда, когда обучающая выборка не содержит объектов с одним и тем же значениям каждого из признаков, принадлежащих разным классам. Однако абсолютная точность на обучающей выборке не всегда приводить к высокой обобщающей способности в результате эффекта переобучения.
Одним из способов достижения более высокой обобщающей способности является использования критериев остановки, позволяющих останови процесс построения дерева до того, как будет достигнута полная однородность концевых вершин.

Решающие деревья. Обучение.

Рассмотри несколько таких критериев.
1. Критерий остановки по минимальному допустимому числу объектов в выборках, соответствующих концевым вершинам.
2. Критерий остановки по минимально допустимой величине индекса $\Delta(\gamma_*, \widetilde{S})$. Предположим, что некоторой вершине ν соответствует выборка $\widetilde{S}_\nu$, для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\widetilde{S}_\nu^l, \widetilde{S}_\nu^r\}$. Вершина ν считается внутренней, если индекс $\Delta(\gamma_*, \widetilde{S})$ превысил пороговое значение τ и считается концевой в противном случае.
3.Критерий остановки по точности на контрольной выборке. Исходная выборка данных случайным образом разбивается на обучающую выборку $\widetilde{S}_t$ и контрольную выборку $\widetilde{S}_c$. Выборка $\widetilde{S}_t$ используется для построения бинарного решающего дерева. Предположим, что некоторой вершине ν соответствует выборка $\widetilde{S}_\nu$, для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\widetilde{S}_\nu^l, \widetilde{S}_\nu^r\}$.

Решающие деревья. Обучение

На контрольной выборке $\widetilde{S}_c$ производится сравнение эффективность распознающей способности деревьев T_ν и T_ν^{++}.
Деревья T_ν и T_ν^{++} включает все вершины и рёбра, построенные до построения вершины ν . В дереве T_ν вершина ν считается концевой. В дереве T_ν^{++} вершина ν считается внутренней, а концевыми считаются вершины, соответствующие подвыборкам $\widetilde{S}_\nu^l$ и $\widetilde{S}_\nu^r$. Распознающая способность деревьев T_ν и T_ν^{++} сравнивается на контрольной выборке $\widetilde{S}_c$. В том, случае если распознающая способность T_ν^{++} превосходит распознающую способность T_ν все дальнейшие построения исходят из того, что вершина ν является концевой. В противном случае производится исследование $\widetilde{S}_\nu^l$ и $\widetilde{S}_\nu^r$.
4. Статистический критерий. Заранее фиксируется пороговый уровень значимости (P<0.05,p<0.01 или p<0.001). Предположим, что нам требуется оценить, является ли концевой вершина , для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\widetilde{S}_\nu^l, \widetilde{S}_\nu^r\}$.

Решающие деревья. Обучение

Исследуется статистическая достоверность различий между содержанием объектов распознаваемых классов в подвыборках $\widetilde{S}_\nu^l$ и $\widetilde{S}_\nu^r$. Для этих целей может быть использованы известные статистический критерий: Хи-квадрат и другие критерии. По выборкам $\widetilde{S}_\nu^l$ и $\widetilde{S}_\nu^r$ рассчитывается статистика критерия и устанавливается соответствующее p-значение. В том случае, если полученное p-значение оказывается меньше заранее фиксированного уровня значимости вершина ν считается внутренней. В противном случае вершина ν считается концевой.

Использование критериев ранней остановки не всегда позволяет адекватно оценить необходимую глубину дерева. Слишком ранняя остановка ветвления может привести к потере информативных предикатов, которые могут быть на самом деле найдены только при достаточно большой глубине ветвления.

Решающие деревья. Подрезка

В связи с этим нередко целесообразным оказывается построение сначала полного дерева, которое затем уменьшается до оптимального с точки зрения достижения максимальной обучающей способности размера путём объединения некоторых концевых вершин. Такой процесс в литературе принято называть «pruning» («подрезка»).
При подрезке дерева может быть использован критерий целесообразности объединения двух вершин, основанный на сравнение на контрольной выборке точности распознавания до и после проведения «подрезки».
Ещё один способ оптимизации обобщающей способности деревьев основан на учёте при «подрезке» дерева до некоторой внутренней вершины ν одновременно увеличения точности разделения классов на обучающей выборке и увеличения сложности, которые возникают благодаря ветвлению из ν.

При этом прирост сложности, связанный с ветвлением из вершины ν, может быть оценён через число листьев в поддереве T_ν^{sub} полного решающего дерева с корневой вершиной ν. Следует отметить, что рост сложности является штрафующим фактором, компенсирующим прирост точности разделения на обучающей выборке с помощью включения поддерева T_ν^{sub} в решающее дерево. Разработан целый ряд эвристических критериев, которые позволяют оценить целесообразность включения T_ν^{sub}. Данные критерии учитывают одновременно сложность и разделяющую способность.

Коллективные решения, ансамбли

Лектор – *Сенько Олег Валентинович*

Курс «Математические методы обучения»

Содержание лекции

Коллективные решения

Задачи прогнозирования некоторой величины Y с помощью предикторов $Z_1, \ldots, Z_M$. Под предикторами понимаются некоторые алгоритмы, прогнозирующие Y. Наборы алгоритмов, над которыми строятся коллективные решения в литературе по машинному обучению принято называть ансамблями. Пусть

$$Z_{ccp} = \sum_{i=1}^{L} c_i Z_i,$$

где

$$\sum_{i=1}^{L} c_i = 1,$$

$$c_i \geq 0, i = 1, \ldots, L$$

Структура ошибки Z_{ccp}

Введём обозначение:
$\delta_i = \mathbb{E}(Y - Z_i)^2 -$ ошибка предиктора Z_i.

$$\sum_{i=1}^{L} \delta_i =$$

$$= \mathbb{E}\{\sum_{i=1}^{L} c_i (Y - Z_{ccp} + Z_{ccp} - Z_i)^2\} =$$

$$= \mathbb{E}\{(Y - Z_{ccp})^2 + \sum_{i=1}^{L} c_i (Z_{ccp} - Z_i)^2 + 2(Y - Z_{ccp}) \sum_{i=1}^{L} (Z_i - Z_{ccp})\}$$

Структура ошибки Z_{ccp}

Однако

$$\sum_{i=1}^{L}(Z_i - Z_{ccp}) = 0$$

по определению Z_{ccp}
Следовательно

$$\delta Z_{ccp} = \mathbb{E}(Y - Z_{ccp})^2 = \sum_{i=1}^{L} \delta_i -$$

$$-\mathbb{E}[\sum_{i=1}^{L} c_i (Z_{ccp} - Z_i)^2]$$

Структура ошибки Z_{ccp}

Рассмотрим слагаемое

$$-\sum_{i=1}^{L} c_i (Z_{ccp} - Z_i)^2$$

Ннетрудно показать, что оно может быть представлено в виде

$$Z_{ccp}^2 - \sum_{i=1}^{L} c_i Z_i^2 = \sum_{i'=1}^{L} \sum_{i''=1}^{L} c_{i'} c_{i''} Z_{i'} Z_{i''} - \sum_{i=1}^{L} c_i Z_i^2$$

Структура ошибки Z_{ccp}

Воспользуемся равенством

$$Z_{i'}Z_{i''} = -\frac{1}{2}\{(Z_{i'} - Z_{i''})^2 -$$

$$-Z_{i'}^2 - Z_{i''}^2\} =$$

Откуда следует

$$\sum_{i'=1}^{L}\sum_{i''=1}^{L} c_{i'}c_{i''}Z_{i'}Z_{i''} - \sum_{i=1}^{L} c_i Z_i^2 =$$

$$= -\frac{1}{2}\sum_{i'=1}^{L}\sum_{i''=1}^{L} c_{i'}c_{i''}(Z_{i'} - Z_{i''})^2 +$$

$$+\frac{1}{2}\sum_{i'=1}^{L} c_{i'}Z_{i'}^2 + \frac{1}{2}\sum_{i''=1}^{L} c_{i''}Z_{i''}^2 - \sum_{i=1}^{L} c_i Z_i^2$$

Структура ошибки Z_{ccp}

Откуда следует

$$-\sum_{i=1}^{L} c_i (Z_{ccp} - Z_i)^2 = -\frac{1}{2} \sum_{i'=1}^{L} \sum_{i''=1}^{L} c_{i'} c_{i''} (Z_{i'} - Z_{i''})^2$$

Введём обозначение

$$\rho_{i'i''} = \mathbb{E}(Z_{i'} - Z_{i''})^2$$

Структура ошибки Z_{ccp}

$$\delta Z_{ccp} = \sum_{i=1}^{L} \delta_i - \frac{1}{2} \sum_{i'=1}^{L} \sum_{i''=1}^{L} c_{i'} c_{i''} \rho_{i' i''}$$

Лекция 9

Структура ошибки выпуклых комбинаций, комитетные методы, логическая коррекция

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Ансамбли алгоритмов

Использование различных методов прогнозирования (распознавания), а также различных обучающих выборок или подмножеств признаков позволяет получить набор прогнозирующих (распознающих) алгоритмов: $A_1, \ldots, A_r$ Можно попытаться увеличить обобщающую способность за счёт выбора алгоритма с минимальной оценкой ошибки прогнозирования. Однако нередко более эффективной процедурой является вычисление прогноза с использованием всех алгоритмов из $A_1, \ldots, A_r$. Использование коллектива (ансамбля) алгоритмов, которые строятся с помощью различных методов позволяет использовать при прогнозировании различные принципы экстраполяции, лежащих в основе этих методов. Статистическое обоснование использованию ансамбля алгоритмов даёт анализ ошибки выпуклой комбинации прогнозов, вычисляемых членами ансамбля. Предположим, что алгоритмы ансамбля $A_1, \ldots, A_r$ вычисляют прогноз переменной Y.

Выпуклые комбинации

Пусть f_i - прогноз, вычисляемый алгоритмом A_i. Тогда

$$\Delta_i = E_\Omega (Y - f_i)^2$$

вляется математическим ожиданием квадрата ошибки прогнозировани для A_i , . Введём обозначение $\rho_{i'i''}$ для математического ожидания квадрата отклонения друг от друга прогнозов, вычисляемых алгоритмами $A_{i'}$ и $A_{i''}$. То есть

$$\rho_{i'i''} = E_\Omega (f_{i'} - f_{i''})^2.$$

Пусть $c_1, \ldots, c_r$ -положительные коэффициенты такие, что $\sum_{i=1}^{r} c_i = 1$ Обозначим через $\hat{f}$ выпуклую комбинацию прогнозов, вычисляемых алгоритмами ансамбля $A_1, \ldots, A_r$. То есть

$$\hat{f} = \sum_{i=1}^{r} c_i f_i.$$

Для ошибки выпуклой комбинации справедливо выражение

$$\hat{\Delta} = E_\Omega(Y - \hat{f})^2 = \sum_{i=1}^{r} c_i \Delta_i - \frac{1}{2} \sum_{i'=1}^{r} \sum_{i''=1}^{r} c_{i'} c_{i''} \rho_{i' i''} \qquad (1)$$

Принимая во внимание, что все квадратичные отклонения $\rho_{i' i''}$ всегда неотрицательны, а коэффициенты $c_1, \ldots, c_r$ положительны получаем неравенство

$$\hat{\Delta} \leq \sum_{i=1}^{r} c_i \Delta_i.$$

Иными словами математическое ожидание квадрата ошибки выпуклой комбинации всегда не превышает аналогичную выпуклую комбинацию математических ожиданий квадратов ошибок отдельных алгоритмов ансамбля.

Рассмотрим, случай, когда все алгоритмы участвуют в построении коллективного решения равноправно. В этом случае $c_i = \frac{1}{r}$, $i = 1, \ldots, r$. Выпуклая комбинация становится просто средним значением

$$\hat{f} = \frac{1}{m} \sum_{i=1}^{r} f_i.$$

Математическое ожидание квадрата ошибки усреднённого по ансамблю прогноза вычисляется по формуле

$$\hat{\Delta} = E_\Omega (Y - \hat{f})^2 = \frac{1}{m} \sum_{i=1}^{r} \Delta_i - \frac{1}{2} \frac{1}{m^2} \sum_{i'=1}^{r} \sum_{i''=1}^{r} c_{i'} c_{i''} \rho_{i' i''} \qquad (2)$$

Таким образом математическое ожидание квадрата ошибки усреднённого по ансамблю прогноза представляет собой разницу между средней по ансамблю величиной математического ожидания квадрата ошибки и средней величиной квадратичного отклонения между прогнозами вычисляемыми различными алгоритмами.

Комитетные методы

Рассмотрим сначала несколько простейших эвристических методов принятия коллективных решений. Предположим, что у нас есть ансамбль алгоритмов распознавания $A_1, \ldots, A_r$, которые были использованы для классификации некоторого объекта s^*.

Голосование по большинству. Простейшим комитетным методом является является метод голосования по большинству, относящий объект к тому классу, к которому он был присвоен относительным большинством алгоритмов.

Использование вещественных оценок за классы.Напомним, что произвольный рспознающий алгоритм является комбинацией распознающего оператора, вычисляющего оценки за классы и решающего правила, производящего классификацию по оценкам, вычисленным распознающим оператором. Предположим, что $\Gamma_l^i(*)$ - оценка за класс , вычисляемая алгоритмом A_i . Коллективное решение может строится путём вычисления коллективных оценок за классы через оценки $\Gamma_l^i(*)$, соответствующие отдельным алгоритмам. При этом могут использоваться различные варианты вычисления

1) Коллективная оценка за класс K_l вычисляется как среднеарифметическое оценок, вычисляемых алгоритмами из ансамбля $\{A_1, \ldots, A_r\}$:

$$\Gamma_l^{av}(s^*) = \sum_{j=1}^{r} \Gamma_l^j(s^*).$$

2) Коллективная оценка за класс K_l вычисляется как вычисляется как минимум всех оценок за данный класс полученных алгоритмами из ансамбля $\{A_1, \ldots, A_r\}$:

$$\Gamma_l^{min}(s^*) = \min_{j \in \{1,\ldots,r\}} \Gamma_l^j(s^*).$$

3) Коллективная оценка за класс K_l вычисляется как вычисляется как максимум всех оценок за данный класс полученных алгоритмами из ансамбля $\{A_1, \ldots, A_r\}$:

$$\Gamma_l^{min}(s^*) = \max_{j \in \{1,\ldots,r\}} \Gamma_l^j(s^*).$$

4) Еще одним употребительным способом построения комитетного решения является использование произведений оценок, вычисляемых алгоритмами из ансамбля $\{A_1, \ldots, A_r\}$:

$$\Gamma_l^{av}(s^*) = \prod_{j=1}^{r} \Gamma_l^j(s^*).$$

К достоинствам комитетных методов относится их простота, высокая быстродействие. Для применения этих методов не требуется никакой дополнительной процедуры обучения, что позволяет сразу переходить к распознаванию объектов комитетом обученных алгоритмов.

Наивный байесовский корректор

Подобными же достоинствами обладает другой известный метод построения коллективных решений – «Наивный байесовский классификатор», который является статистическим методом, основанном на оценках вероятностей принадлежности объекта классам в зависимости от результатов классификации отдельными алгоритмами. Предположим, что алгоритмы $\{A_1, \ldots, A_r\}$ отнесли объект s^*в классы $K_{J(1)}, \ldots, K_{J(r)}$ соответственно. Факт отнесения объекта s в класс K_i алгоритмом A_j далее будем обозначать $A_j(s) = \mathrm{Pt}_i(s)$, где $\mathrm{Pt}_i(s)$ является предикатом, обозначающим отнесение s в класс K_i. Наибольшую точность распознавания обеспечивает байесовский классификатор, относящий объект в класс K_{i^*}, для которого максимальной является условная вероятность

$$P[s^* \in K_{i^*} \mid A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(r)}(s^*)] \qquad (3)$$

Условная вероятность (1) для класса K_i может быть вычислена по формуле Байеса

Наивный байесовский корректор

$$P[s^* \in K_i \mid A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(r)}(s^*)] =$$

$$= \frac{P(K_i)P[A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(r)}(s^*) \mid s^* \in K_i]}{P[A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(1)}(s^*)]}.$$

Условная вероятность

$$P[A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(1)}(s^*) \mid s^* \in K_i]$$

может быть оценена исходя из гипотезы о независимости входящих в ансамбль классификаторов. То есть

$$P[A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(r)}(s^*) \mid s^* \in K_i] =$$

$$= \prod_{j=1}^{r} P[A_1(s^*) = \mathrm{Pt}_{J(j)}(s^*) \mid s^* \in K_i].$$

Наивный байесовский корректор

В качестве оценок вероятностей

$$P(K_1), \ldots, P(K_l)$$

$$P[A_1(s^*) = \mathrm{Pt}_{J(j)}(s^*) \mid s^* \in K_i],$$

при

$$j = 1, \ldots, r, i = 1, \ldots, r$$

могут быть использованы соответствующие доли объектов обучающей выборки. Отметим, что вероятность

$$P[A_1(s^*) = \mathrm{Pt}_{J(1)}(s^*), \ldots, A_r(s^*) = \mathrm{Pt}_{J(r)}(s^*)]$$

является одинаковым для всех классов множителей, который может не учитываться при вычислении окончательного решения.

Логическая коррекция

Комитетные методы и наивный байесовский классификатор являются простейшими методами коллективной коррекции, не учитывающих взаимодействие алгоритмов ансамбля или их относительную эффективность. Требование повышения обобщающей способности ансамбля за счёт более полного учёта его структуры и использования возможностей лежащих в его основе эвристик. привело к созданию средств алгебраической и логической коррекции. Методы логической коррекции учитывают только окончательные результаты классификации. Пусть у нас имеется некоторая выборка $\widetilde{S}_c = \{s_1, \ldots, s_q\}$ объектов, принадлежащих классам $K_1, \ldots, K_L$, по которой мы собираемся произвести коррекцию. Данной выборке может быть сопоставлена информационная матрица $\|\alpha_{li}\|_{L \times q}$, , где $\alpha_{li} = 1$ при $s_i \in K_l$ и $\alpha_{ij} = 0$ в противном случае.

.

Логическая коррекция

Выборке $\widetilde{S}_c$ может быть также сопоставлен набор матриц

$$\{\|\beta_{li}^j\|_{L\times q} \mid j = 1, \ldots, r\}.$$

Элемент $\beta_{li}^j = 1$, если $A_j(s_i) = \mathrm{Pt}_l(s_i)$, и $\beta_{li}^j = 0$ в противном случае. Поиск оптимального логического корректора сводится к поиску для каждого класса такой логической функции $F_l(z_1, \ldots, z_r)$ от булевых переменных $z_1, \ldots, z_r$, чтобы равенство

$$F_l[\beta_{li}^{g(1)}, \ldots, \beta_{li}^{g(r)}] = \alpha_{li}$$

выполнялось для возможно большего числа объектов выборки $\widetilde{S}_c$. Функция $g(i)$ устанавливает связь между переменными $z_1, \ldots, z_r$ и алгоритмами $A_1, \ldots, A_r$. Использование g позволяет учитывать информативность алгоритмов для оценки принадлежности распознаваемых объектов классу K_l, через их место в логической функции.

Логическая коррекция

Предположим, что отсутствуют противоречия типа существования в выборке $\widetilde{S}_c$ объектов $s_{i'}$ и $s_{i''}$ с неодинаковыми $\alpha_{li'}$ и $\alpha_{li''}$, которые однако одинаково классифицируются алгоритмами ансамбля. То есть $\beta^j_{li'} = \beta^j_{li''}$ при $j = 1, \ldots, r$.
В случае, если задана какая-либо функция g, а множество векторов

$$\{(\beta^1_{li}, \ldots, \beta^r_{li}) \mid i = 1, \ldots, r\}$$

включает всё множество вершин единичного куба $\mathbb{E}^r$, логическая функция F_l оказывается полностью определённой. В противном случае задача построения логического корректора включает в себя задачу доопределения логической функции естественным путём заданной на выборке $\widetilde{S}_c$ на весь единичный куб $\mathbb{E}^r$.

Логическая коррекция

Одним из способов логической корреции является построение монотонных корректоров, которое сводится к поиску такой функции g, что логическая функция F_l, правильно вычислющая элнменты информационной матрицы, является монотонной. То есть ищется функция $g(i)$, которая

а) удовлетворяет равенству

$$F_l[\beta_{li}^{g(1)}, \ldots, \beta_{li}^{g(r)}] = \alpha_{li}$$

при $i = 1, \ldots, q$;

б) для любых векторов $(z'_1, \ldots, z'_r)$ и $(z''_1, \ldots, z''_r)$, удовлетворяющих условию

$$(z'_1, \ldots, z'_r) \succeq (z''_1, \ldots, z''_r)$$

выполняется неравенство

$$F_l(z'_1, \ldots, z'_r) \succeq F_l(z''_1, \ldots, z''_r)$$

.

Логическая коррекция

Построение монотонных корректоров сводится к следующей схеме. В исходном наборе $A_1, \ldots, A_r$ для каждого класса K_l выбирается поднабор $A_{f(1)}, \ldots, A_{f(k)}$. Объект s относится монотонным логическим корректором в класс K_l в том и только в том случае, если он отнесён в K_l всеми алгоритмами из $A_{f(1)}, \ldots, A_{f(k)}$ и ещё одним алгоритмом из набора $A_1, \ldots, A_r$, который не принадлежит $A_{f(1)}, \ldots, A_{f(k)}$.

Алгебраическая коррекция

Универсальным способом построения оптимального распознающего алгоритма по набору исходных алгоритмов $A_1, \ldots, A_r$ является использование алгебраических методов коррекции. В отличие от логических методов коррекции алгебраические методы используют не только окончательные результаты классификации, содержащиеся в матрицах $\|\beta_{li}^j\|_{L\times q}$, но также матрицы оценок $\|\gamma_{li}^j\|_{L\times q}$, вычисляемые операторами $R_1, \ldots, R_k$, входящими в алгоритмы $A_1, \ldots, A_r$. Элемент γ_{li}^j является оценкой объекта s_i за класс K_l , вычисляемая оператором R_j, $i = 1, \ldots, q$,$l = 1, \ldots, L$, $j = 1, \ldots, r$. Основы теории алгебраической коррекции были разработаны Ю.И.Журавлёвым в 1976-1978 годах. Задача распознавания в алгебраической теории рассматривается как задача построения по начальной информации I о классах $K_1, \ldots, K_L$ для предъявленной для распознавания выборки $\widetilde{S}_c = \{s_1, \ldots, s_q\}$ правильной информационной матрицы $\|\alpha_{li}^j\|_{L\times q}$. .

Алгебраическая коррекция

Последнюю задачу мы будем называть задачей $\mathrm{Z}(I, \widetilde{S}_c, \mathrm{Pt}_1, \ldots, \mathrm{Pt}_L)$ или просто задачей Z. Примером начальной информации о классах является таблица признаковых описаний эталонных объектов классов и их информационная матрица. Предположим, что у нас имеется множество алгоритмов $\{A\}$, переводящих пару $(I, \widetilde{S}_c)$ в матрицы $\|\beta_{li}^j\|_{L\times q}$, составленные из элементов $\{0, 1, \Delta\}$, где значения 1 и 0 как и раньше соотвествуют истинности или ложности предикатов, вычисленными алгоритмами из множества $\{A\}$, значение Δ соответствует отказу от вычисления значения предиката.

Определение 1. Алгоритм A называется корректным для задачи Z, если выполнено равенство

$$A(I, \widetilde{S}_c, \mathrm{Pt}_1, \ldots, \mathrm{Pt}_L) = \|\alpha_{li}\|_{L\times q}.$$

Алгоритм, не являющийся корректным для задачи Z , называется некорректным.

Алгебраическая коррекция

Совокупность $\{A\}$ состоит из вообще говоря некорректных алгоритмов. Алгебраический подход к решению задач распознавания включает в себя введение алгебраических операций над алгоритмами из $\{A\}$, позволяющих строить корректные алгоритмы по наборам алгоритмов из $\{A\}$. Поскольку каждый распознающий алгоритм может быть представлен как последовательное выполнение распознающего оператора и решающего правила, множеству $\{A\}$ соответствуют множества операторов $\{R\}$ и множество решающих правил $\{C\}$. Каждый из операторов из множества $\{R\}$ вычисляет для задачи Z матрицу оценок за классы

$$R^*(I, \widetilde{S}_c) = \|\gamma_{li}^*\|_{L \times q}$$

На множестве операторов $\{R\}$ вводятся операции сложения, умножения и умножения на скаляр.

Алгебраическая коррекция

Предположим, что R' и R'' являются операторами из $\{R\}$. При этом $R'(I, \widetilde{S}_c) = \|\gamma'_{li}\|_{L\times q}$ и $R''(I, \widetilde{S}_c) = \|\gamma''_{li}\|_{L\times q}$. Пусть b является некоторой скалярной величиной. Операция умножения на скаляр преобразует оператор R' в оператор $(b \bullet R')$, задаваемый формулой

$$(b \bullet R')(I, \widetilde{S}_c) = \|b\gamma'_{li}\|_{L\times q}, \tag{4}$$

Сумма операторов $(R' + R'')$ задаётся формулой

$$(R' + R'')(I, \widetilde{S}_c) = \|\gamma'_{li} + \gamma''_{li}\|_{L\times q}, \tag{5}$$

Произведение операторов $(R' \bullet R'')$ задаётся формулой

$$(R' + R'')(I, \widetilde{S}_c) = \|\gamma'_{li} \cdot \gamma''_{li}\|_{L\times q}, \tag{6}$$

Алгебраическая коррекция

Использование операций (4)-(6) позволяет строить новые распознающие операторы, являющиеся полиномами от операторов из исходного множества вида

$$\sum_{i=1}^{N_p} a_i [R_{t(1,i)} \bullet \ldots \bullet R_{t(k(i),i)}]$$

Функция $t(j,i)$ указывает на оператор, находящийся в позиции j слагаемом с номером i, k_i -число сомножителей в слагаемом с номером i. Очевидно, что замыкание $\mathbf{L}\{R\}$ множества операторов $\{R\}$ относительно операций (4) и (5) является линейным векторным пространством. Обозначим через $\mathbf{U}\{R\}$ алгебраическое замыкание множества $\{R\}$ относительно операций (4)-(6).
Рассмотрим условия, существования корректного алгоритма для некоторой задачи $\mathrm{Z}(I, \widetilde{S}_c, \mathrm{Pt}_1, \ldots, \mathrm{Pt}_L)$.

. **Определение 2.** Если множество матриц $\{R(I, \widetilde{S}_c)\}$, где операторы пробегают множество $\{R\}$, содержит базис в пространстве числовых матриц размерности $L \times q$, то задача $\mathrm{Z}(I, \widetilde{S}_c, \mathrm{Pt}_1, \ldots, \mathrm{Pt}_L)$ называется полной относительно $\{R\}$.

Определение 3. Решающее правило C называется корректным , если для всякой выборки длины q существует хотя бы одна числовая матрица $\|\gamma'_{li}\|_{L\times q}$ такая, что

$$C(\|\gamma'_{li}\|_{L\times q}) = \|\alpha_{li}\|_{L\times q}$$

Пусть $\{A\}$ является множество алгоритмов вида $A = R \otimes C^*$, где $R \in \{R\}$, C^* - некоторое корректное решающее правило.

Определение 4. Множества алгоритмов вида $A = R \otimes C^*$, будут обозначаться $\mathbf{L}\{A\}$ и $\mathbf{U}\{A\}$, если $R \in \mathbf{L}\{R\}$ и $R \in \mathbf{U}\{R\}$ соответственно.

Алгебраическая коррекция

Пусть C^* - некоторое корректное решающее правило.

Теорема 1. Если множество $\{\mathrm{Z}\}$ состоит лишь из задач , полных относительно $\{R\}$, то линейное замыкание $\mathbf{L}\{A = R \otimes C^*\}$, является корректным относительно $\{\mathrm{Z}\}$.

Доказательство. Пусть M является матрицей, которая может быть переведена решающим правилом C^* в информационную матрицу $\|\alpha_{li}\|_{L\times q}$. Существование матрицы M следует из корректности решающего правила C^*. При фиксированном q базис в пространстве числовых матриц размерности $L \times q$ состоит из Lq матриц $M_1, \ldots, M_{Lq}$.Тогда существуют такие числа $c_1, \ldots, c_{Lq}$, что $M = \sum_{i=1}^{Lq} c_i M_i$.

В том случае, если матрицы $M_1, \ldots, M_{Lq}$ построены из $\{I, \widetilde{S}_c\}$ с помощью операторов $R_1, \ldots, R_{Lq}$ из $\{R\}$, корректный алгоритм A_{corr} можетбыть представлен в виде

$$A_{corr} = \sum_{i=1}^{Lq} c_i R_i \otimes C^*$$

Теорема доказана.

Следствие 1. Пусть $\{A\}$ - совокупность некорректных алгоритмов, $\{R\}$ - соответствующее множество операторов, C^* - некоторое фиксированное корректное решающее правило. Тогда $\mathbf{L}\{A = R \otimes C^*\}$ является корректным относительно $\{\mathrm{Z}\}$, если $\{\mathrm{Z}\}$ состоит из задач, полных относительно $\{R\}$.

Следствие 2. Пусть выполнены все условия следствия 1. Тогда $\mathbf{U}\{A\}$ является корректным относительно $\{\mathrm{Z}\}$, если $\{\mathrm{Z}\}$ состоит из задач, полных относительно замыкания $\mathbf{U}\{A\}$.

Линейные и алгебраические замыкания могут строится не только над конечными наборами заранее обученных алгоритмов, но также и над множествами алгоритмов, принадлежащих некоторой модели и имеющих в общем случае мощность континуума. Рассмотрим в качестве примера рассмотрим один из вариантов модели алгоритмов вычисления оценок, в котором оценки за класс K_l вычисляются по формуле

$$\Gamma_l(s^*) = \sum_{s_\mu \in K_l} \sum_{\omega \in \Omega} \Big(\sum_{j=1}^{n} p_j \omega_j\Big) B_\omega(s^*, s_\mu, \varepsilon) + \tag{7}$$

$$+ \sum_{s_\mu \in \widetilde{S} \setminus K_l} \sum_{\omega \in \Omega} \Big(\sum_{j=1}^{n} p_j \omega_j\Big) \bar{B}_\omega(s^*, s_\mu, \varepsilon)$$

Алгебраическая коррекция

В формуле (7) используются следующие обозначения

- $\bar{B}_{\boldsymbol{\omega}}(s^*, s_\mu, \boldsymbol{\varepsilon}) = 1 - B_{\boldsymbol{\omega}}(s^*, s_\mu, \boldsymbol{\varepsilon})$ является функцией антиблизости объекта к эталону по опорному множеству, описываемому бинарным характеристическим вектором $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$;
- $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)$- вектор положительных пороговых коэффициентов, задающих близость объектов по каждому из признаков;
- $(p_1, \ldots, p_n)$-вектор положительных параметров, характеризующих важность признаков,
- $(\gamma_1, \ldots, \gamma_n)$ вектор положительных параметров, характеризующих важность объектов.

Алгебраическая коррекция

Для того, чтобы описать условия существования корректного алгоритма в алгебраическом замыкании подмножества алгоритмов вычисления оценок введём дополнительные определения. Пусть $\widetilde{M}$ - некоторое множество допустимых объектов.

Определение 5. Объекты s_u и s_ν с описаниями $(x_1^u, \ldots, x_n^u)$ и $(x_1^\nu, \ldots, x_n^\nu)$ называются изоморфными относительно множества $\widetilde{M}$, если для произвольного объекта s с описанием $(a_1, \ldots, a_n) \in \widetilde{M}$ выполняются равенство

$$(\mid x_1^u - a_1 \mid, \ldots, \mid x_n^u - a_n \mid) = (\mid x_1^\nu - a_1 \mid, \ldots, \mid x_n^\nu - a_n \mid)$$

Нетрудно видеть что корректный алгоритм в рамках модели вычисления оценок по формуле (7) не может существовать для задачи $\mathrm{Z}(I, \widetilde{S}_c, \mathrm{Pt}_1, \ldots, \mathrm{Pt}_L)$ случаях, когда оказываются выполненными следующие два условия.

Алгебраическая коррекция

- в выборке $\widetilde{S}_c$ существует два объекта s' и s'' , изоморфных относительно выборки эталонов $\widetilde{S}_e$, которая вместе со своей информационной матрицей образует начальную информацию I;
- объекты s' и s'' принадлежат двум непересекающимся классам. Действительно в этом случае векторы оценок $[\Gamma_1(s'),\ldots,\Gamma_L(s')]$ и $[\Gamma_1(s''),\ldots,\Gamma_L(s'')]$, вычисляемые произвольным оператором из модели АВО будут одинаковы. Следовательно никакое множество операторов модели АВО не может вычислять базис в пространстве вещественных матриц размера $L \times q$.

Будем называть задачу $\mathrm{Z}(I,\widetilde{S}_c,\mathrm{Pt}_1,\ldots,\mathrm{Pt}_L)$ регулярной при выполнении трёх условий:

- никакие два класса полностью не совпадают, т.е. $K_{l'} \neq K_{l''}$ при $l' \neq l''$;
- никакие два объекта из $\widetilde{S}_c$ не являются изоморфными относительно выборки эталонов $\widetilde{S}_e$;
- $\widetilde{S}_e \bigcap \widetilde{S}_c = \varnothing$

Алгебраическая коррекция

Справедлива следующая теорема.

Теорема 2. Алгебраическое замыкание подкласса алгоритмов модели ABO, в которой оценки за классы вычисляются по формуле (5) корректно над множеством регулярных задач.

Лекция 10

Коллективные метод, бэггинг, бустинг, голосование по системам закономерностей

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Коллективные методы ("бэггинг")

Одним из способов получения ансамбля является использование алгоритмов, обученных по разным обучающим выборкам, которые возникают в результате случайного процесса, лежащего в основе исследуемой задачи. Обычно при решении прикладной задачи в распоряжении исследователя имеется обучающая выборка $\widetilde{S}_t = \{s_1, \ldots, s_m\}$ ограниченного объёма. Однако процесс генерации семейства выборок из генеральной совокупности может быть имитирован с помощью процедуры бутстрэп (bootstrap), которая основана на выборках с возвращениями из $\widetilde{S}_t$. В результате получаются выборки $\widetilde{S}_*^{bg}$, включающие объекты из обучающей выборки $\widetilde{S}_t$. Однако некоторые объекты $\widetilde{S}_t$ могут встречаться в $\widetilde{S}_*^{bg}$ более одного раза, а другие объекты отсутствовать. Предположим, что с помощью процедуры бутстрэп получено T выборок. С помощью заранее выбранного метода, используемого для обучения отдельных алгоритмов распознавания, получим множество, включающее T распознающих алгоритмов $\widetilde{A}_{bg} = \{A_1^{bg}, \ldots, A_T^{bg}\}$.

Коллективные методы "бэггинг"

Для получения коллективного решения может быть использован простейший комитетный метод, относящий объект в тот класс, куда его отнесло большинство алгоритмов . Данная процедура носит название бэггинг (bagging), что является сокращением названия Bootstrap Aggregating. Процедура бэггинг показывает высокий прирост обобщающей способности по сравнению с алгоритмом, обученным с помощью базового метода по исходной обучающей выборке $\widetilde{S}_t$, в тех случаях, когда вариационная составляющая ошибки базового метода высока. К таким моделям относятся в частности решающие деревья и нейросетевые методы. При использовании в качестве базового метода решающих деревьев процедура бэггинг приводит к построению ансамблей решающих деревьев (решающих лесов).
Основной идеей алгоритма бустинг является пошаговое наращивание ансамбля алгоритмов. Алгоритм, который присоединяется к ансамблю на шаге k обучается по выборке, которая формируется из объектов исходной обучающей выборки $\widetilde{S}_t$.

Коллективные методы "бустинг"

В отличие от метода бэггинг объекты выбираются не равноправно, а исходя из некоторого вероятностного распределения, заданного на выборке $\widetilde{S}_t$. Данное распределение вычисляется по результатам классификации с помощью ансамбля, полученного на предыдущем шаге. Приведём схему одного из наиболее популярных вариантов метода бустинг AdaBoost (Adaptive boosting) более подробно. На первом шаге присваиваем начальные значения весов $(w_1^1, \ldots, w_m^1)$ объектам обучающей выборки. Поскольку веса имеют вероятностную интерпретаци, то для них соблюдаются ограничения $\sum_{j=1}^m w_j^1 = 1$, $w_j^1 \in [0, 1]$. Обычно начальное распределение выбирается равномерным $w_j^1 = \frac{1}{m}$, $j = 1, \ldots, m$. Выбираем число итераций T. На итерации k генерируем выборку $\widetilde{S}_k^{bs}$ из исходной выборки $\widetilde{S}_t$ согласно распределению задаваемому весами $(w_1^k, \ldots, w_m^k)$. Обучаем распознающий алгоритм A_k^{bs} по выборке $\widetilde{S}_k^{bs}$.

Вычисляем взвешенную ошибку по формуле $\varepsilon_k = \sum_{j=1}^{m} w_j^k e_j^k$, где $e_j^k = 1$, если алгоритм A_k^{bs} неправильно классифицировал объект s_j и $e_j^k = 0$ в противном случае. В том случае, если $\varepsilon_k \geq 0.5$ или $\varepsilon_k = 0$ игнорируем шаг и заново генерируем выборку $\widetilde{S}_k^{bs}$ исходя из весовых коэффициентов $w_j^1 = \frac{1}{m}$, $j = 1, \ldots, m$. В случае если $\varepsilon_k \in (0, 0.5)$ вычисляем коэффициенты $\tau_k = \frac{\varepsilon_k}{1-\varepsilon_k}$
и пересчитываем веса объектов по формуле

$$w_j^{k+1} = \frac{w_j^k (\tau_k)^{1-e_j^k}}{\sum_{j=1}^{m} w_j^k (\tau_k)^{1-e_j^k}} \tag{1}$$

при $j = 1, \ldots, m$.
Процесс, задаваемый формулой (1), продолжается до тех пор, пока не выполнено T итераций . В результате мы получаем совокупность из T распознающих алгоритмов $A_1^{bs}, \ldots, A_T^{bs}$.

Предположим, что нам требуется распознать объект s^*. Пусть $\beta_l^k(s^*) = 1$, если s^* отнесён алгоритмом A_k^{bs} в класс K_l, и $\beta_l^k(s^*) = 0$ в противном случае. Оценка объекта s^* за класс K_l вычисляется по формуле

$$\Gamma_l(s^*) = \sum_{k=1}^{T} \ln \frac{1}{\tau_k} \beta_l^k(s^*).$$

Объект s^* будет отнесён к классу, оценка за которой максимальна. Описанный вариант метода носит название AdaBoost. M1. Эффективность процедур бустинга подтверждается многочисленными экспериментами на реальных данных. В настоящее время существует большое количество вариантов метода, имеющих разное обоснование.

Коллективные методы, основанные на голосовании по системам закономерностей

Одним из эффективных подходов к решению задач прогнозирования и распознавания является использование коллективных решений по системам закономерностей. Под закономерностью понимается распознающий или прогностический алгоритм, определённый на некоторой подобласти признакового пространства или связанный с некоторым подмножеством признаков. В качестве примера закономерностей могут быть приведены представительные наборы, являющиеся по сути подмножествами признаковых описаний, характерных для одного из распознаваемых классов. Аналогом представительный наборов в задач с веществennозначной информацией являются логические закономерности классов. Под логической закономерностью класса K_l понимается область признакового пространства, имеющая форму гиперпараллелепипеда и содержащая только объекты из K_l.

Логическая закономерность

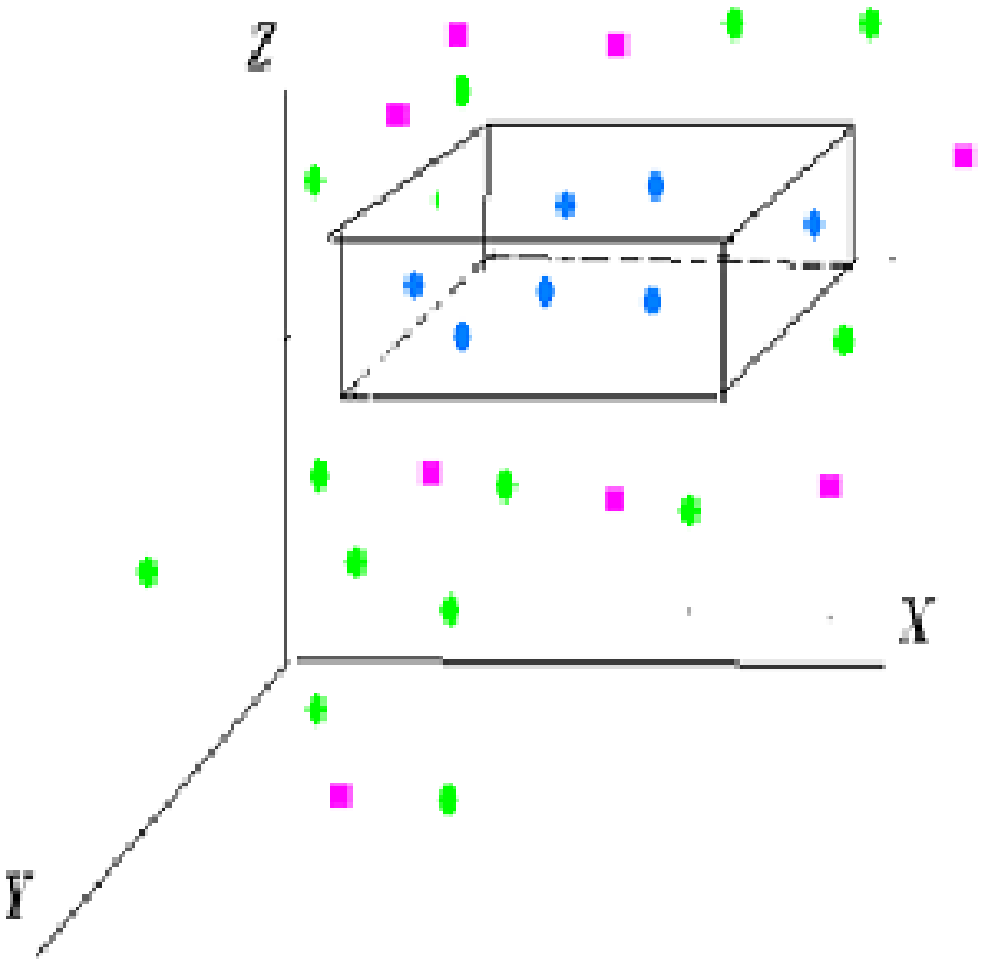


Рис 1. Пример логической закономерности.

Логическая закономерность

Логическая закономерность класса K_j, которую обозначим $\boldsymbol{r}(j)$, описывается с помощью предикатов вида

$$Pt_i^{\boldsymbol{r(j)}} = \text{”}a_i^{\boldsymbol{r}(j)} \leq X_i \leq b_i^{\boldsymbol{r}(j)} \tag{2}$$

где $i = 1, \ldots, n$. Отметим, что для несущественных для закономерности $\boldsymbol{r}(j)$ признаков отрезок $[a_i^{\boldsymbol{r}(j)}, b_i^{\boldsymbol{r}(j)}]$ соответствует области допустимых значений X_i. Для существенных признаков отрезок $[a_i^{\boldsymbol{r}(j)}, b_i^{\boldsymbol{r}(j)}]$ является некоторым подмножеством области допустимых значений X_i. Полностью $\boldsymbol{r}(j)$ задаётся конъюнкцией предикатов (2):

$$\mathrm{Pt}^{\boldsymbol{r}(j)} = Pt_i^{\boldsymbol{r(j)}} \& \ldots \& Pt_n^{\boldsymbol{r(j)}}.$$

Логическая закономерность

Для конъюнкции $\mathrm{Pt}^{\boldsymbol{r}(j)}$ должны выполняться следующие условия:

- в обучающей выборке $\widetilde{S}_t$ должен существовать объект s^* из класса K_j, для которого $\mathrm{Pt}^{\boldsymbol{r}(j)} = 1$;
- в обучающей выборке $\widetilde{S}_t$ не должно содержаться объектов, не принадлежащих класса K_j, для которых $\mathrm{Pt}^{\boldsymbol{r}(j)} = 1$;
- $\mathrm{Pt}^{\boldsymbol{r}(j)}$ доставляет экстремум некоторому функционалу качества $\Phi(\mathrm{Pt})$, заданному на множестве всевозможных предикатов, удовлетворяющих условиям 1), 2)

На практике используются следующие функционалы качества:

- число объектов из класса K_j в обучающей выборке, для которых $\mathrm{Pt}^{\boldsymbol{r}(j)} = 1$;
- доля объектов из класса K_j в обучающей выборке, для которых $\mathrm{Pt}^{\boldsymbol{r}(j)} = 1$;

Логическая закономерность

Наряду с полными логическими закономерностями, для которых выполняются все условия 1) – 3), используются также частичные логические закономерности, для которых допускаются некоторые нарушения условия 2). То есть допускается существование небольшой доли нарушений условия 2) для тех объектов, для которых выполняется условие $\mathrm{Pt}^{\boldsymbol{r}(j)} = 1$. На этапе обучения для каждого из классов K_j ищется множество логических закономерностей $\widetilde{R}_j$. Предположим, что нам требуется распознать новый объект s^* . Для каждого из классов K_j ищется число закономерностей из $\widetilde{R}_j$ для которых $\mathrm{Pt}^{\boldsymbol{r}(j)}(s^*) = 1$. При этом доля таких закономерностей считается оценкой за класс K_j . Для классификации используется стандартное решающее правило, т.е. объект относится в класс, оценка за который максимальна. Поиск оптимальной системы логических закономерностей производится по набору $\widetilde{S}_e$ случайно выбранных из обучающей выборки $\widetilde{S}_t$ эталонных объектов (опорных эталонов).

Логическая закономерность

Закономерности для класса K_j строится по каждому из опорных эталонов $s_i \in \widetilde{S}_e$. При этом поиск оптимальных границ

$$[a_1^{\boldsymbol{r}(j)}, b_1^{\boldsymbol{r}(j)}, \ldots, a_n^{\boldsymbol{r}(j)}, b_n^{\boldsymbol{r}(j)}]$$

для закономерности $\boldsymbol{r}(j)$ осуществляется сначала на некоторой неравномерной сетке пространства, которая задается с помощью разбиения интервала значений каждого из признаков. После нахождения оптимальных границ на заданной сетке, поиск продолжается на заданной в окрестности этого оптимального решения, но уже на более мелкой сетке. Процесс заканчивается, если при переходе к более мелкой сетке не удается найти логическую закономерность с более высоким критерия качества Φ. . Задача поиска оптимальной логической закономерности на каждой сетке сводится к поиску максимальной совместной подсистемы некоторой системы неравенств. Логические закономерности, построенные для случайно выбранных «опорных» эталонов класса K_j объединяются в одно множество $\widetilde{R}_j$.

Статистически взвешенные синдромы

Коллективные решения в методе СВС принимается по информации о принадлежности векторного описания распознаваемого объекта так называемым “синдромам” из некоторого множества $\widetilde{Q}$. Под "синдромом"понимается такая область признакового пространства, в которой содержание объектов одного из классов, отличается от содержания объектов этого класса в обучающей выборке или по крайней мере в одной из соседних областях. Пример синдромов, характеризующих разделение объектов из классов K_1 (+) и K_2 (0) приведён на рисунке 2. Синдромы ищутся для каждого из распознаваемых классов с помощью построения оптимальных разбиений интервалов допустимых значений единичных признаков или совместных двумерных областей допустимых значений пар признаков.

Статистически взвешенные синдромы

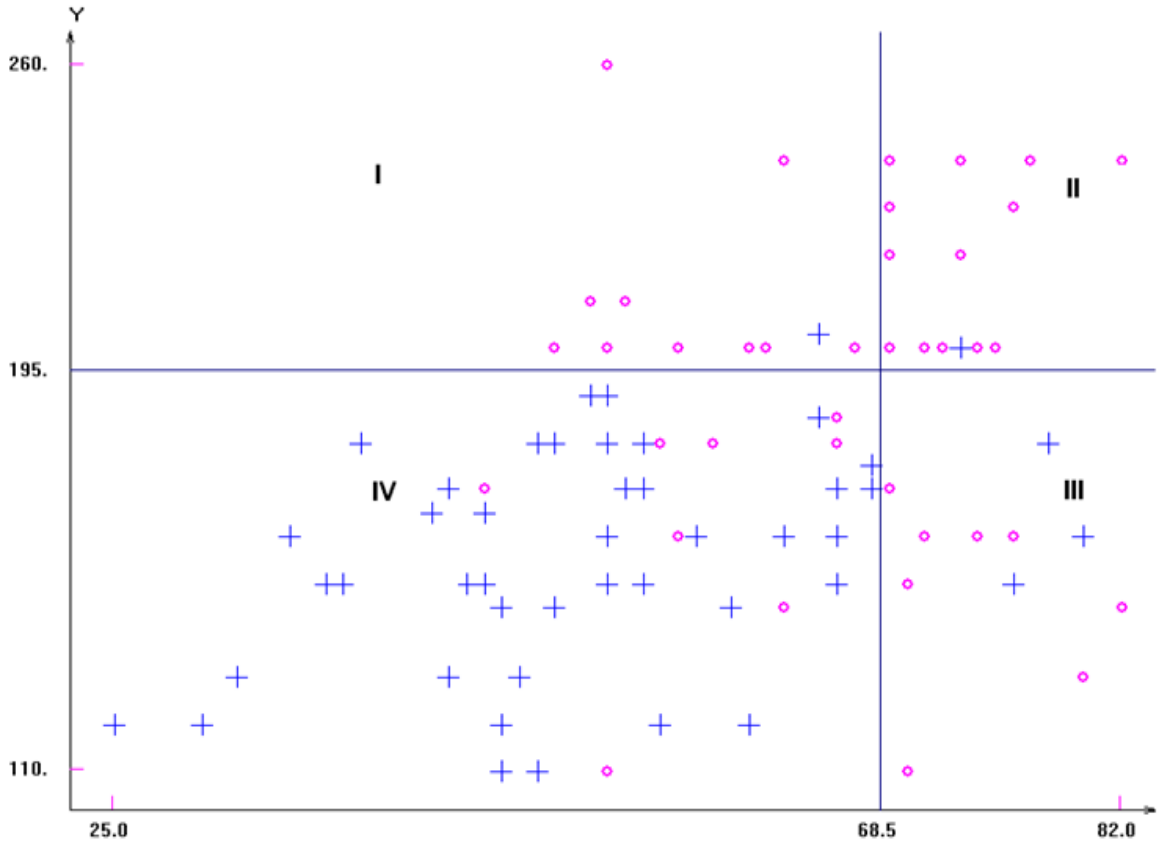


Рис 2. Видно, описания объектов из сосредоточены главным образом в нижнем левом квадранте «синдроме».

Статистически взвешенные синдромы

При этом поиск производится внутри нескольких семейств разбиений, имеющих различный уровень сложности. В ходе поиска выбирается разбиение с максимальным значением функционала качества. Используется два функционала качества, зависящих от обучающей выборки $\widetilde{S}_t$, распознаваемого класса K_l, и разбиения R:

- интегральный $F_i(\widetilde{S}_t, K_l, R)$;
- локальный $F_{loc}(\widetilde{S}_t, K_l, R)$.

Обозначим через $q_1, \ldots, q_r$ элементы некоторого разбиения R . Пусть ν_0^l является долей объектов класса K_l в обучающей выборке $\widetilde{S}_t$, ν_i^l - доля объектов K_l среди объектов, описания которых принадлежат элементу q_i, m_i- число объектов, описания которых принадлежат элементу q_i. Интегральный функционал определяется формулой

$$F_i(\widetilde{S}_t, K_l, R) = \sum_{i=1}^{r} (\nu_0^l - \nu_i^l)^2 m_i.$$

Статистически взвешенные синдромы

Локальный функционал определяется формулой

$$F_i(\widetilde{S}_t, K_l, R) = \max_{i=1,\ldots,r} (\nu_0^l - \nu_i^l)^2 m_i.$$

Поиск разбиений с максимальным значением одного из функционалов производится в рамках одного из четырёх семейств. Примеры разбиений для каждого из семейств приведены на рисунке.
Семейство I включает всевозможные разбиения интервалов допустимых значений отдельных признаков на два интервала с помощью одной граничной точки.
Семейство II включает всевозможные разбиения интервалов допустимых значений отдельных признаков на 3 интервала с помощью двух граничных точек.

Статистически взвешенные синдромы

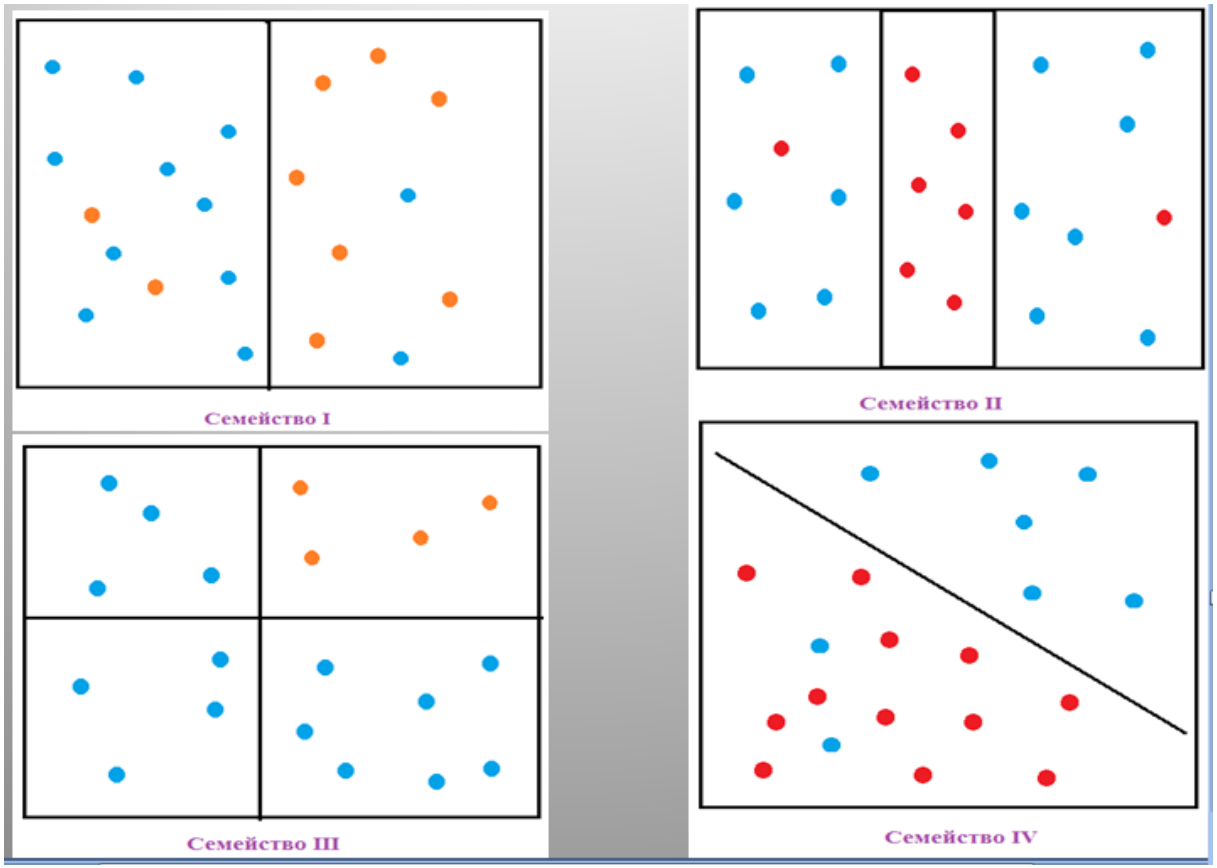


Рис 3. Примеры разбиений для каждого из четырёх семейств, используемых в методе CBC.

Статистически взвешенные синдромы

Семейство III включает всевозможные разбиения совместных двумерных областей допустимых значений пар признаков на 4 подобласти с помощью двух граничных точек (по одной точке для каждого из двух признаков).
Семейство IV включает всевозможные разбиения совместных двумерных областей допустимых значений пар признаков на 2 подобласти с помощью прямой граничной линии, произвольно ориентированной относительно координатных осей.
Найденные оптимальные разбиения используются для построения систем синдромов, если соответствующая им максимальная величина функционала качества превосходит некоторое заранее заданное пользователем пороговое значение δ. Причём величина порога зависит от сложности модели разбиений. Порог является минимальным для простейшей одномерной модели I. Для моделей II-IV величина порога домножается на величину , задаваемую пользователем, что позволяет регулировать влияние эффекта переобучения.

Статистически взвешенные синдромы

Одномерные разбиения, найденные внутри семейств I и II могут быть используются при построении не только одномерных, но также и двумерных синдромов. Предположим, что на этапе обучения для класса K_l найдена система синдромов $\widetilde{Q}_l$. Предположим, что описание $\boldsymbol{x}^*$ распознаваемого объекта s^* принадлежит синдромам $q_1, \ldots, q_r$ из системы $\widetilde{Q}_l$. Оценка s^* за класс K_l вычисляется по формуле

$$\Gamma_l(s^*) = \frac{\sum_{i=1}^{r} w_i^l \nu_i^l}{\sum_{i=1}^{r} w_i^l},$$

где ν_i^l - доля класса K_l в синдроме q_i, w_i^l - вес синдрома при классификации класса K_l. Вес синдрома вычисляется по формуле

$$w_i^l = \frac{m_i}{m_i + 1} \frac{1}{\nu_i^l(1 - \nu_i^l)},$$

где m_i - число объектов обучающей выборки с описанием, принадлежащем q_i.

Метод комитетов

Метод комитетов представляет собой реализацию подхода к решению задач распознавания, объединяющего принципы линейного разделения классов и вычисления коллективных решений. Рассмотрим задачу распознавания с двумя классами K_1 и K_2 . Пусть $\widetilde{f} = \{f_1(\boldsymbol{x}), \ldots, f_r(\boldsymbol{x})\}$ является набором линейных функций вида

$$f_i(\boldsymbol{x}) = a_{1i}x_1 + \ldots + a_{ni}x_n,$$

где $\boldsymbol{x} = (x_1, \ldots, x_n)$ является вектор используемых для распознавания признаков, $(a_{1i}, \ldots, a_{ni})$- вектор вещественных параметров, задающих линейную функцию $f_i(\boldsymbol{x})$. Каждая из функций из $\widetilde{f}$ рассматривается в качестве отдельного линейного классификатора, относящего объект с описанием $\boldsymbol{x}$ в класс K_1, если $sign[f_i\boldsymbol{x}] > 0$, и в класс K_2 в противном случае. .

Метод комитетов

Предположим, что для классификации произвольного объекта s с описанием $\boldsymbol{x}$ используется следующее решающее правило метода комитетов:

- объект s относится в класс K_1 , если $\sum_{i=1}^{r} sign[f_i(\boldsymbol{x})] > 0$;
- объект s с описанием $\boldsymbol{x}$ относится в класс K_2 , если $\sum_{i=1}^{r} sign[f_i(\boldsymbol{x})] < 0$;
- в случае, если $\sum_{i=1}^{r} sign[f_i(\boldsymbol{x})] = 0$ происходит отказ от распознавания.

Набор функций $\widetilde{f}$ называется комитетом, если решающее правило метода комитетов правильно классифицирует объекты обучающей выборки.

Метод, основанный на поиске комитетов, потенциально позволяет производить распознавание линейно неразделимых классов, реализуя кусочно-линейную разделяющую поверхность. Обучение сводится к поиску оптимальных (минимальных по числу функций) комитетов. Теоретически показано существование комитета для непротиворечивых данных.

Лекция 12

Байесовские сети
Методы анализа выживаемости

Лектор – *Сенько Олег Валентинович*

Курс «Математические основы теории прогнозирования»
4-й курс, III поток

Содержание лекции

Байесовские сети

Рассмотренные ранее в курсе методы позволяют прогнозировать значения отдельных целевых переменных. Однако более высокая точность прогноза для сложных технических, биологических или социальных систем может быть достигнута на основе описания взаимодействия наборов переменных, характеризующих данные системы. Подробное наглядное описание взаимодействия больших наборов переменных может быть достигнуто с использованием современных графических вероятностных моделей. К числу подобных моделей относятся байсовские сети (БС), сочетающие графическую наглядность с математической строгостью. Аппарат байесовских сетей принципиально позволяет полностью охарактеризовать многомерное совместное распределение больших наборов переменных.

Определение 1. Байесовской сетью называется ориентированный ациклический граф, вершинам которого поставлены в соответствие случайные переменные.

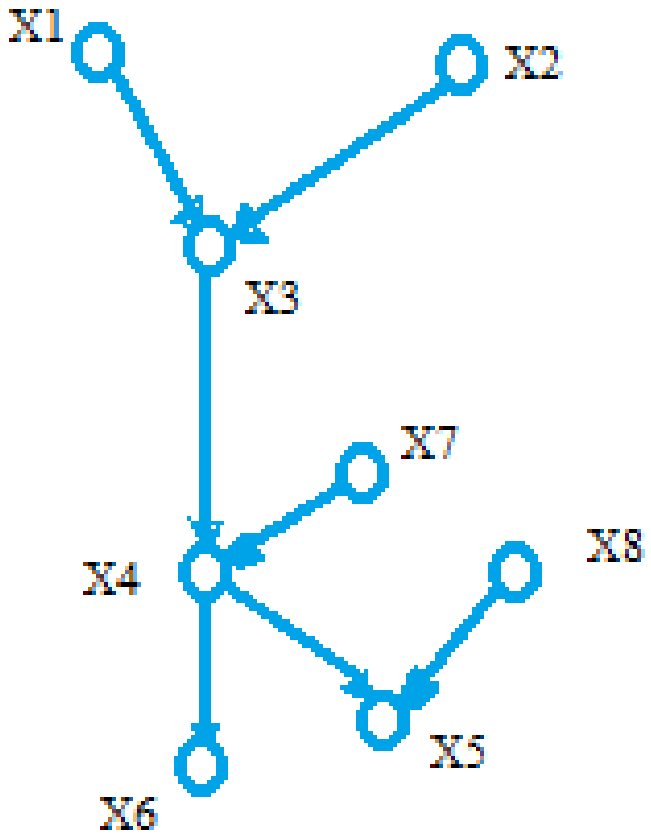


Рис.1. Пример байесовской сети.

При этом наличие ребра между двумя вершинами указывает на наличие статистической связи между соответствующими переменными. Иногда направление ребра интерпретируют как наличие причинно-следственной связи между соответствующими переменными. Для того, чтобы более точно охарактеризовать смысловую связь графической структуры БС с совместными распределениями наборов переменных, введём дополнительные определения. Вершина X_i называется предком вершины X_j, если они соединены ребром, ориентированным от X_i к X_j. Соответственно вершина X_j считается потомком вершины X_i. Вершины, не имеющие предком, называются корневыми. Обозначим через $Par(X)$ – множество предков вершины X, $V^i(X)$ - множество вершин БС, не являющихся потомками вершины X и не содержащее вершину X. Вершина называется корневой, если у неё нет вершин предков. Пример БС, описывающей взаимосвязь переменных $X_1, \ldots, X_8$, приведён на рисунке 1. Вершины, соответствующие переменным X_1, X_2, X_7, X_8 являются корневыми. Вершины, соответствующие переменным X_1 и X_2 являются предками вершины, соответствующей переменной X_3.

Вершины, соответстющие переменным X_3 и X_7, являются предками вершины, соответствующей переменной X_4 и т.д.

Условие 1. Байесовская сеть строится исходя из требования об условной независимости каждой вершины X от множества вершин $V^i(X)$ при известных значениях родителей из $Par(X)$.

Можно показать, что выполнение условия 1 эквивалентно справедливости разложения для совместного распределения вершин $X_1, \ldots, X_n$:

Условие 2.

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P[X_i | Pa(X_i)]. \quad (1)$$

Из условия 2 видно, что совместная вероятность $P(X_1, \ldots, X_n)$ может быть описана с помощью n условных распределений $P[X_i|Par(X_i)]$.

Предположим, что переменные $X_1, \ldots, X_n$ являются категориальными. Тогда о параметры распределений $P[X_i|Par(X_i)]$ задаются с помощью таблицы условных вероятностей (ТУВ).

Ячейки ТУВ, соответствующей вершине (переменной) X_i, содержат вероятности каждого из возможных значений X_i при всевозможных комбинациях значений узлов, являющихся родителями X_i. В случае, когда БС является разреженной, (т.е. когда число связей между вершинами оказывается существенно ниже общего числа парных сочетаний вершин), суммарный объём ТУВ оказывается существенно меньше общего числа всевозможных комбинаций значений переменных$(X_1, \ldots, X_n)$. Построении БС производится по обучающей выборке, содержащей значения векторов переменных $X_1, \ldots, X_n$, измеренные, например, в различные моменты времени. Для оценки условной независимости используются статистические тесты. Обучающие выборки используются для вычисления ТУВ. Вместе с тем задание общего каркаса БС часто производится экспертом в области знаний,где используется БС. Построенная БС может быть использована для решения нескольких типов задач. В первую очередь необходимо отметить задачу вероятностного вывода.

Целью вероятностного вывода является оценка вероятности состояний каждой из вершин сети, исходя из известных значений переменных, соответствующих корневым вершинам. Для расчётов может быть использована представление совместной вероятности условия 2. Предположим, что переменные $X_1, \ldots, X_8$ являются бинарнымии, принимающими значения из множества $\{0, 1\}$. Рассчитаем вероятность $X_6 = 1$ при следующих условиях, наложенных на корневые переменных: $X_1 = 0$,$X_2 = 0$,$X_7 = 1$,$X_8 = 1$. Для этого достаточно вычислить, использую формулу (1) вероятность каждой из комбинаций значений переменных вида $(0, 0, u_3, u_4, u_5, u_6, 1, 1)$, где u_3, u_4, u_5, u_6 выбираются из множества $\{0, 1\}$. Обозначим множество таких комбинаций через $\widetilde{U}_f$. Разобъём множество $\widetilde{U}_f$ на подмножества $\widetilde{U}_0$, включающие комбинации из $\widetilde{U}_f$ с $u_6 = 0$, и $\widetilde{U}_1$. , включающие комбинации из $\widetilde{U}_f$ с $u_6 = 1$.

Очевидно, что вероятность множества комбинаций $\widetilde{U}$ является суммой вероятностей комбинаций, входящих в $\widetilde{U}$. Вероятность $X_6 = 1$ при условии $X_1 = 0, X_2 = 0, X_7 = 1, X_8 = 1$ очевидно равна

$$P\{\widetilde{U}_1|\widetilde{U}_f\} = \frac{P(\widetilde{U}_1)}{P(\widetilde{U}_f)}.$$

При моделировании с помощью БС сложных технических систем корневым вершинам соответствуют переменные, характеризующие внешние воздействия, или управляющие параметры. Процедура вероятностного вывода позволяет оценить распределения вероятностей для переменных, характеризующих возникновение нарушений функционирования технической системы при заданных внешних воздействиях в зависимости от значений набора управляющих параметров.

Анализ выживаемости

Ранее нами рассматривались разнообразные средства решения задачи распознавания и задачи прогнозирования непрерывных переменных (регрессионного анализа). Однако в различных прикладных исследованиях и практической деятельности встречаются задачи, которые не могут быть адекватно решены только лишь с помощью данных средств. К числу таких задач следует отнести задачу анализа выживаемости в медицине и биологии или задачу анализа надёжности в технике. Целью таких задач является восстановление вероятности того, что ожидаемое критическое событие с исследуемым объектом произойдёт не ранее произвольного момента времени. Таким критическим событием может быть отказ изделия в технике, гибель испытуемого организма в биологии или смерть пациента в медицине. Таким образом целью анализа является вычисление функции (кривой) выживаемости $S(t) = P\{T > t\}$, где через T обозначено время наступления критического события, $P\{T > t\}$ обозначает вероятность того, что критическое событие произойдёт позже момента t.

Обычно момент t отсчитывается от от некоторой важной для изучаемого процесса точки. Такой точкой может быть, например, момент производства изделия или момент начала лечения. Следует отметить, что в большинстве практических исследованиях важно не только вычислить кривую выживаемости, но и оценить влияние на неё переменных, характеризующих исследуемые объекты. Такими переменными могут быть, например, возраст пациента и различные клинические показатели в биомедицинских исследованиях, или параметры, характеризующие условия изготовления изделия, в задачах анализа надёжности.

Задача расчёта кривых выживаемости и оценки влияния на них различных переменных может быть решена с помощью методов моделирования по эмпирическим данным. Методы анализа выживаемости по эмпирическим данным тесно связаны с цензурированностью информации. Наблюдение в статистике считается цензурированным, если известно не точное значение наблюдаемой величины, а только интервал, которому оно принадлежит.

Данный интервал может быть как конечным, так и бесконечным (ограниченным с одной стороны). В данных, связанных с анализом выживаемости или надёжности нередко цензурированной оказывается информация о наступлении критического события. Например, в анализируемой выборке может содержаться информация не только об объектах, для которых критическое событие уже наступило, и момент этого события был точно зафиксирован, но также и об объектах, для которых критическое событие на момент последнего наблюдения не произошло. Выборки данных в задачах анализа выживаемости обычно имеют вид

$$\widetilde{S} = \{s_1 = (\alpha_1, t_1, \boldsymbol{x}_1), \ldots, s_m = (\alpha_m, t_m, \boldsymbol{x}_m)\},$$

где t_i - время, прошедшее от начального момента до момента последнего наблюдения за объектом;
α_i - индикатор, равный 1, если в момент t_i для объекта s_i было зафиксировано критическое событие, и равный , если в момент критическое событие не наступило;
$\boldsymbol{x}_i = (x_{i1}, \ldots, x_{in})$- вектор переменных $X_1, \ldots, X_n$, которые потенциально могут оказывать влияние на форму кривой выживаемости

Анализ выживаемости

Рассмотрим методы восстановления кривых выживаемости при игнорировании влияния на их форму переменных $X_1, \ldots, X_n$ Одним из наиболее популярных методов восстановления кривых выживаемости в этих случаях является процедура Каплан-Майера, учитывающая существование цензурированных наблюдений. При отсутствии таких наблюдений процедура Каплан-Майера эквивалентна вычислению обычных эмпирических наблюдений. Предположим, что наблюдения в некоторой выборке $\widetilde{S}$ фиксировались в моменты $t_1, \ldots, t_N$. Пусть n_i-число объектов, для которых критический момент не наступил до момента времени t_i, d_i -число критических событий в момент t_i. Оценка значения кривой выживаемости мо методу Каплан-Майера на полуинтервале $(t_i, t_{i+1}]$ вычисляется по формуле

$$S(t) = \prod_{j=1}^{i} \frac{n_j - d_j}{n_j}.$$

На рисунке 1 представлены примеры оценок кривых выживаемости по методу Каплан-Майера.

Анализ выживаемости

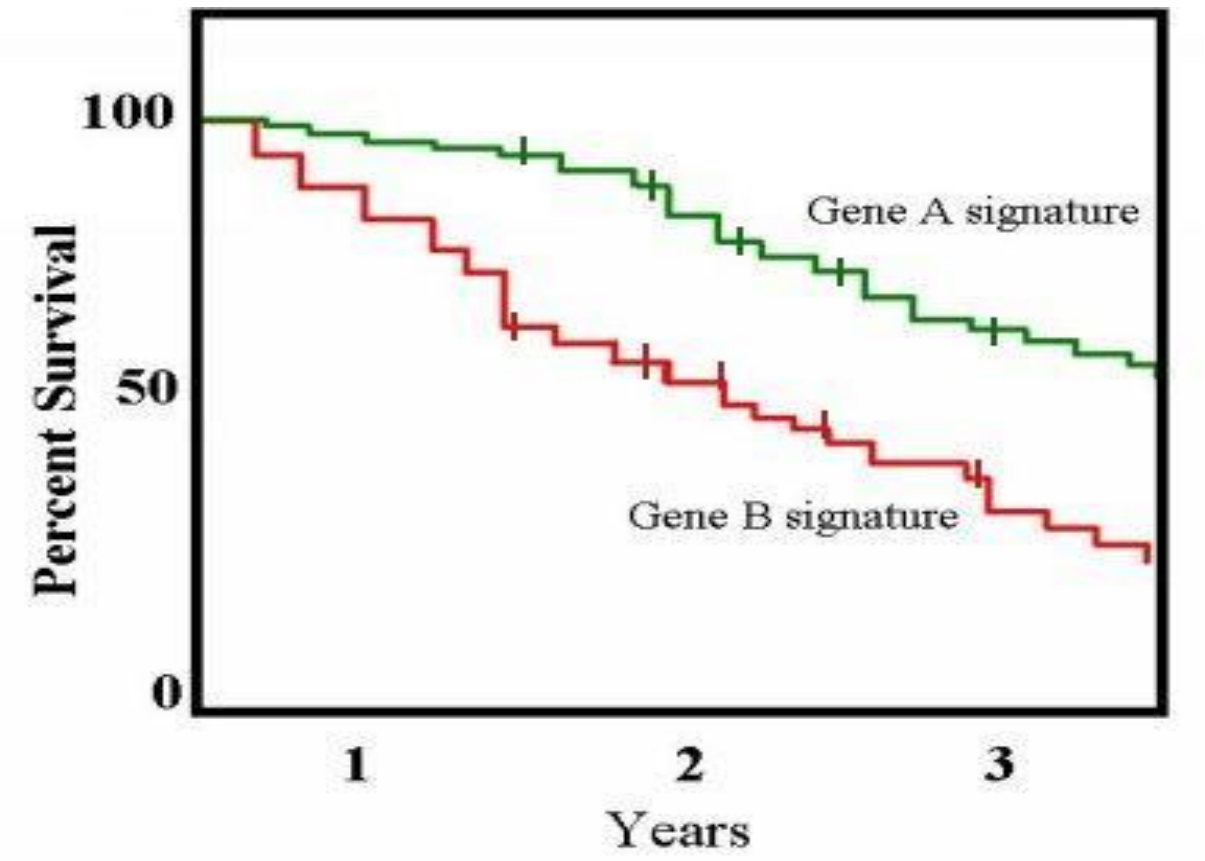


Рис. 1. Сравниваются оценки для кривых выживаемости по методу Каплан-Майера групп пациентов с двумя вариантами генотипа.

В настоящее время существует целый ряд методов оценки влияния переменных $X_1, \ldots, X_n$ на форму кривой выживаемости. Одной из популярных моделей до сих пор является модель Кокса, основанная на концепции мгновенного риска. Мгновенный риск $\lambda(t)$ в момент t определяется как предел

$$\lim_{\Delta t \to 0} = \frac{P[T \leq (t + \Delta t) | T \geq t]}{\Delta t} = \frac{f(t)}{S(t)},$$

где $f(t)$ плотностью вероятности наступления критического события в точке t. То есть $f(t) = \frac{dF(t)}{dt}$, где $F(t) = 1 - S(t)$. Таким образом очевидна справедливость простого дифференциального уравнения

$$\lambda(t)dt = -\frac{-dS(t)}{S(t)}. \tag{2}$$

Проинтегрировав левую и правую части уравнения (1) на отрезке $[t_0, t]$ убеждаемся в справедливости равенств
$\ln[S(t)] = -\Lambda(t)$ или $S(t) = \exp[-\Lambda(t)]$ где $\Lambda(t) = \int_{t_0}^{t} \lambda(t)$.

В случае если форма кривой выживаемости зависит от переменных $X_1, \ldots, X_n$, мгновенный риск также оказывается функцией переменных $X_1, \ldots, X_n$. В основе модели Кокса (модели пропорциональных рисков) лежит предположение о возможности представления мгновенного риска для произвольного объекта s_* с описанием $\boldsymbol{x}_* = (x_1^*, \ldots, x_n^*)$ в виде произведения

$$\lambda(t|\boldsymbol{x}_*) = \lambda_0(t) \exp\left(\beta_1 * x_1^* + \ldots + \beta_n * x_n^*\right),$$

где $\lambda_0(t)$ - базовая компонента, зависящая только от времени. Пусть $S_0(t) = \exp[-\Lambda_0(t)]$, где $\Lambda_0(t) = \int_{t_0}^{t} \lambda_0(t)$. В результате получаем

$$S(t) = S_0(t)^{\left[\exp\left(\beta_1 * x_1^* + \ldots + \beta_n * x_n^*\right)\right]}.$$

Для поиска вектора параметров $(\beta_1, \ldots, \beta_n)$ используется метод максимального правдоподобия.

Модель пропорциональных рисков Кокса

Предположим, что для настройки модели пропорциональных рисков используется обучающая выборка $\widetilde{S} = \{s_1 = (\alpha_1, t_1, \boldsymbol{x}_1), \ldots, s_m = (\alpha_m, t_m, \boldsymbol{x}_m)\}$. Предположим, что критическое событие для объекта s_i произошло в момент времени t_i. Вероятность того, что среди всех объектов, для которых критическое событие до момента t_i не наступало, это событие в момент t_i произошло именно с s_i оценим с помощью отношения

$$\frac{\lambda(t_i|\boldsymbol{x}_i)}{\sum_{t_j>t_i}\lambda(t_i|\boldsymbol{x}_j)} = \frac{\lambda_0(t_i)\exp\left(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in}\right)}{\sum_{t_j>t_i}\lambda_0(t_i)\exp\left(\beta_1 * x_{j1} + \ldots + \beta_n * x_{jn}\right)} =$$

$$= \frac{\exp\left(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in}\right)}{\sum_{t_j>t_i}\exp\left(\beta_1 * x_{j1} + \ldots + \beta_n * x_{jn}\right)}$$

Функционал правдоподобия записывается в виде

$$L(\beta_1, \ldots, \beta_n) = \prod_{i=1}^{m} \frac{\exp\left(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in}\right)}{\sum_{t_j > t_i} \exp\left(\beta_1 * x_{j1} + \ldots + \beta_n * x_{jn}\right)}.$$

В модели используются значения $(\beta_1, \ldots, \beta_n)$, при которых $L(\beta_1, \ldots, \beta_n)$ достигает максимума. Наряду со значением параметров $(\beta_1, \ldots, \beta_n)$ неизвестным параметром модели пропорциональных рисков является форма базовой функции выживаемости $S_0(t)$. Одним из возможных способов восстановления $S_0(t)$ является подход, основанный на аппроксимация отношения

$$\frac{S(t_i|\beta_1, \ldots, \beta_n, \boldsymbol{x}_i)}{S(t_{i-1}|\beta_1, \ldots, \beta_n, \boldsymbol{x}_i)}$$

величиной

$$1 - \frac{\exp\left(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in}\right)}{\sum_{t_j > t_i} \exp\left(\beta_1 * x_{j1} + \ldots + \beta_n * x_{jn}\right)} \tag{3}$$

для произвольной пары последовательных моментов времени (t_{i-1}, t_i), для которых имели место критические события.

При этом предполагается, что вектор параметров $(\beta_1, \ldots, \beta_n)$ уже был ранее найден с помощью описанного ранее варианта метода максимального правдоподобия. Очевидно, что для вектора $\boldsymbol{x}_i$, описывающего объект s_i из обучающей выборки, справедливо равенство

$$\frac{S(t_i|\beta_1, \ldots, \beta_n, \boldsymbol{x}_i)}{S(t_{i-1}|\beta_1, \ldots, \beta_n, \boldsymbol{x}_i)} = [\frac{S_0(t_i)}{S_0(t_{i-1})}]^{exp(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in})}. \quad (4)$$

Обозначим отношение $\frac{S_0(t_i)}{S_0(t_{i-1})}$ через γ_i. Из равенств (2) и (3) следует справедливость равенства

$$\gamma_i = [1 - \frac{\exp(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in})}{\sum_{t_j > t_i} \exp(\beta_1 * x_{j1} + \ldots + \beta_n * x_{jn})}]^{[exp(\beta_1 * x_{i1} + \ldots + \beta_n * x_{in})]^{-1}}$$

Очевидно, величина γ_i может быть рассчитана для каждого объекта из выборки$\widetilde{}$.

Оценка базовой функции выживаемости на отрезке времени $[t_i, t_{i+1}]$ может оцениваться в виде произведения коэффициентов γ_i по всевозможным объектам $\widetilde{S}$, для которых критическое событие наступило до момента t_i. То есть

$$S_0(t_i) = \prod_{t_j < t_i} \gamma_j.$$

Временные ряды

Под временным рядом понимается множество значений некоторой переменной Z, измеренных в моменты времени, разделённые одинаковыми интервалами

$$\ldots, Z(t_{i-1}), Z(t_i), Z(t_{i+1}), \ldots$$

Временной ряд считается многомерным, если в каждый момент времени измеряются значения нескольких переменных. Многомерный ряд, содержащий значения переменных $Z_1, \ldots, Z_k$, может быть представлен в виде набора последовательностей:

$$\ldots, Z_1(t_{i-1}), Z_1(t_i), Z_1(t_{i+1}), \ldots$$

$$\ldots, \ldots, \ldots, \ldots, \ldots, \ldots$$

$$\ldots, Z_k(t_{i-1}), Z_k(t_i), Z_k(t_{i+1}), \ldots$$

Временные ряды

Основной задачей анализа временных рядов является поиск алгоритма, позволяющего предсказывать значения переменной Z или значения переменных из некоторого подмножества $Z_1, \ldots, Z_k$ в ещё не наступившие моменты времени. Дополнительными задачами анализ временных рядов является поиск существующих эмпирических закономерностей, включая поиск циклических изменений переменных. Прогнозирование временного ряда производится с помощью алгоритма, обученного по доступному в результате наблюдений участку временного ряда достаточной длины. Одним из способов прогнозирования временных рядов является использование одномерной регрессионной функции $f(t)$, зависящей от времени. В тех случаях, когда прогностическая способность $f(t)$ является статистически достоверной, а функция $f(t)$ является линейной, говорят о наличии во временном ряду линейного тренда. Для поиска линейного тренда может быть использован метод простой одномерной регрессии с использованием в качестве прогнозирующей переменной X время t.

Значения переменной Z в различных точках временного ряда

$$\ldots, Z(t_{i-1}), Z(t_i), Z(t_{i+1}), \ldots$$

могут рассматриваться как реализации случайных функций

$$\ldots, \breve{Z}_{i-1}, \breve{Z}_i, \breve{Z}_{i+1}, \ldots.$$

Процесс, отображаемый временным рядом, называется стационарным, если совместное распределение вероятности для произвольных r последовательно расположенных в ряду случайных величин

$$\breve{Z}_{i+1}, \ldots, \breve{Z}_{i+r}$$

Совпадает с совместным распределением r случайных величин

$$\breve{Z}_{i+1+l}, \ldots, \breve{Z}_{i+r+l}, \ldots$$

при некотором целом l.

Очевидно, что процесс является стационарным, если переменные

$$\ldots, \breve{Z}_{i-1}, \breve{Z}_i, \breve{Z}_{i+1}, \ldots$$

являются независимыми и одинаково распределёнными.
Предположим, что функция $f(t)$ полностью характеризует процесс. Это означает, что $Z(t_i) = f(t_i) - \varepsilon_i$, где $\ldots, \varepsilon_{i-1}, \varepsilon_i, \varepsilon_{i+1}, \ldots$ - независимые и одинаково распределённые ошибки с нулевым математическим ожиданием. Тогда случайный процесс, отображаемый временным рядо

$$\ldots, [Z(t_{i-1}) - f(t_{i-1})], [Z(t_i) - f(t_i)], [Z(t_{i+1}) - f(t_{i+1})], \ldots,$$

оказывается стационарным.

Для прогнозирования временного ряда в произвольной точке t_i наряду с методами, основанными на выделении тренда, используются методы, основанные на поиске оптимального алгоритма A , вычисляющего оценку $Z(t_i)$ по набору предшествующих значений $\{Z(t_{j_1}), \ldots, Z(t_{j_l})\}$, где $(j_1, \ldots, j_l)$ является набором целых чисел. То есть оценка $\hat{Z}(t_i)$ вычисляется по формуле

$$\hat{Z}(t_i) = A[Z(t_{j_1}), \ldots, Z(t_{j_l})].$$

Простейшим примером такого рода прогнозирования является метод скользящего среднего, вычисляющего оценку $\hat{Z}(t_i)$ в виде

$$\hat{Z}(t_i) = \frac{1}{l} \sum_{j=1}^{l} Z(t_{i-j}).$$

Используется также метод взвешенного скользящего среднего, вычисляющего оценку $\hat{Z}(t_i)$ в виде

$$\hat{Z}(t_i) = \frac{1}{l}\sum_{j=1}^{l} c_j Z(t_{i-j}),$$

где $(c_1, \ldots, c_l)$ являются неотрицательными коэффициентами, удолетворяющими условию $\sum_{j=1}^{l} c_j = 1$.
Нетрудно видеть, что прогностическая способность метода скользящего связана с относительным постоянство математического ожидания случайных величин $\breve{Z}_{i-1}, \ldots, \breve{Z}_{i-l}, \ldots$. Метод скользящего среднего используется для "сглаживания" временных рядов, фильтрации высокочастотной шумовой составляющей.

В общем случае для обучения алгоритма A могут быть использованы всевозможные методы регрессионного анализа и распознавания, если прогнозируемая переменная Z является категориальной. Обучение алгоритма A может производится по таблице, составленный из элементов, принадлежащих известному участку временного ряда. Предположим, что в результате наблюдений стали известны значения $Z(t_1), \ldots, Z(t_N)$. По данному ряду может быть построена таблица

$$Z(t_N), Z(t_{N-1}), \ldots, Z(t_{N-l}),$$
$$Z(t_{N-1}), Z(t_{N-2}), \ldots, Z(t_{N-l-1}),$$
$$\ldots, \ldots, \ldots, \ldots, \ldots, \ldots,$$
$$Z(t_{N-l}), Z(t_{N-l-1}), \ldots, Z(t_{N-2l}).$$

При этом первый слева элемент в каждой строке рассматривается в качестве прогнозируемой величины Y . Далее последовательно слева направо значения переменной Z в строке рассматриваются в качестве значений прогнозирующих переменных $X_1, \ldots, X_l$. В случае многомерных временных рядов при прогнозировании некоторой переменной Z_j могут быть использованы значения и других переменных из набора $Z_1, \ldots, Z_k$. .

Для поиска циклических (сезонных) колебаний переменной Z могут быть использованы методы корреляционного анализа. Для каждой предполагаемой длины цикла l строится таблица, состоящая из двух столбцов:

$$Z(t_N), Z(t_{N-l}),$$

$$Z(t_{N-1}), Z(t_{N-l-1}),$$

$$\ldots, \ldots, \ldots$$

$$Z(t_{l+1}), Z(t_1).$$

Вычисляется коэффициента корреляции между столбцами. Реально существующему циклу длины l^* соответствует максимальная величина коэффициента корреляции для таблицы, построенной по сдвигу l^*, по отношению к коэффициентам корреляции для таблиц, построенным исходя из других величин сдвига.

2Кластерный анализ

Целью методов кластерного анализа является разбиение выборок многомерных данных на группы объектов близких в смысле некоторой заданной меры сходства. Такие компактные группы называются кластерами, классами или таксонами.

Методы кластерного анализа называют также методами обучения без учителя, автоматической группировки или таксономии.

Методы кластерного анализа могут использоваться в качества вспомогательных инструментов при решении задач прогнозирования или распознавания. Однако нередко кластеризация может иметь самостоятельное значение.

2 Кластерный анализ

Большинство известных алгоритмов кластеризации предполагает задание расстояния $\rho(\mathbf{x},\mathbf{y})$ между произвольными векторами-описаниями

объектов. В качестве расстояния могут выступать, например, евклидова метрика:

$$\rho(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Используются и другие функции расстояния.

2Кластерный анализ

Одним из наиболее известных методов кластеризации является **алгоритм k внутригрупповых средних**. Предположим, что у нас задана выборка векторов- описаний $\tilde{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$. Алгоритм находит такие кластеры, для объектов которых центр «своего кластера» будет ближе центра любого «чужого кластера».

Метод предполагает, что число кластеров изначально задано.

2Кластерный анализ

Поиск оптимальной кластеризации методом *к внутригрупповых* средних

Предположим, что предполагаемое число кластеров равно ***r.***

Зададим произвольным образом исходное разбиение выборки $\tilde{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ на группы $G_1^0, \ldots, G_r^0$

Вычисляем геометрические центры исходных группПусть группа G_i^0 состоит из объектов $\{\mathbf{x}_1^0, \ldots, \mathbf{x}_{m(i)}^0\}$. Тогда центр G_i^0 вычисляется по формуле $\overline{\mathbf{x}}_i^0 = \frac{1}{m(i)} \sum_{j=1}^{m(i)} \mathbf{x}_j^0$

Вычисляются расстояния между объектами из $\tilde{S}$ и центрами $\overline{\mathbf{x}}_1^0, \ldots, \overline{\mathbf{x}}_r^0$

2Кластерный анализ

Поиск оптимальной кластеризации методом *k* внутригрупповых средних

Объекты из $\tilde{S}$ затем переносятся в группу с наименее удалённым центром. В результате мы получаем новый набор групп $G_1^1,\ldots,G_r^1$. Повторяем для набора групп $G_1^1,\ldots,G_r^1$ те же самые операции, которые ранее выполнялись для групп $G_1^0,\ldots,G_r^0$

……

Процесс завершается на некотором шаге ***k+1***, когда переносы объектов из $\tilde{S}$ в другие группы не требуются.

То есть каждый объект наименее удалён от центра той же самой группы, которой он и принадлежит. В результате мы получаем набор компактных групп - кластеров

Иерархическая кластеризация

Для того, чтобы осуществить иерархическую кластеризацию необходимо сначала задать расстояние $\mathrm{P}(G',G'')$ между произвольными кластерами G',G''.

Возможные способы задания расстояния:

1) $\mathrm{P}(G',G'') = \min_{\mathbf{x}'\in G',\mathbf{x}''\in G''} \rho(\mathbf{x}',\mathbf{x}'')$ - то есть расстоянием между двумя кластерами является минимальное расстояние между двумя объектами, один из которых принадлежит G', а второй G''.

2) $\mathrm{P}(G',G'') = \max_{\mathbf{x}'\in G',\mathbf{x}''\in G''} \rho(\mathbf{x}',\mathbf{x}'')$ - то есть расстоянием между двумя кластерами является максимальное расстояние между двумя объектами, один из которых принадлежит G', а второй G''.

Иерархическая кластеризация

3) $\mathrm{P}(G', G'') = \rho(\bar{\mathbf{x}}', \bar{\mathbf{x}}'')$ - расстояние между центрами кластеров G', G''

4) $\mathrm{P}(G', G'') = \frac{1}{m'm''} \sum_{i=1}^{m'} \sum_{j=1}^{m''} \rho(\mathbf{x}'_i, \mathbf{x}''_j)$ - среднее расстояние между объектами из двух кластеров

Отметим, что в случае, когда все кластеры состоят только из одного объекта, расстояния между ними всегда равны расстояниям между этими единственными объектами.

Иерархическая кластеризация

На начальном этапе кластерами являются объекты $\tilde{S}$

На каждом последующем шаге происходит объединение двух ближайших кластеров из набора, образованного на предыдущем шаге.

Процесс завершается при достижении одного из следующих условий.

1) Кластеры, образованные на новом шаге теряют компактность. Тогда мы оставляем в силе кластеризацию, полученную на предыдущем шаге.
2) Образуется требуемое число кластеров
3) Процесс завершается, если достигнутая кластеризация удовлетворяет требованиям эксперта исследователя.

ИССЛЕДОВАНИЯ ФОЛЬКЛОРНО-МИФОЛОГИЧЕСКИХ ТРАДИЦИЙ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

Целью настоящей работы является разработка и обоснование методов интеллектуального анализа данных, эффективных при исследовании фольклорно-мифологических традиций по наборам представленных в них мотивов. База данных, содержащая информацию о встречаемости мотивов, создана и Ю.Е. Березкиным [Березкин 2007; 2009] и размещена на сайте http://starling.rinet.ru/kozmin/tales/index.php?index=berezkin

В 2007 г. база включала сведения о встречаемости 1355 мифологических мотивов в 337 традициях (на ноябрь 2009 в ней 1483 мотива и 470 традиций). Для этого на протяжении почти двадцати лет были проанализированы более 5500 публикаций на германских, романских, славянских и прибалтийско-финских языках, использованы также некоторые неопубликованные материалы. Под мотивом понимаются повторяющиеся образы, эпизоды или их сочетания максимальной протяженности, встречающиеся в двух и более (практически - во многих) текстах. В базу данных включались только такие мотивы, которые обнаружены не менее, чем в четырех традициях. Под традицией понимается совокупность текстов, записанных у одной этно-языковой группе.

В базе данных для всех традиций в бинарной форме фиксируется наличие или отсутствие каждого мотива в проанализированных источниках.

Следует подчеркнуть, что наличие 0 в некоторой позиции традиции не обязательно достоверно свидетельствует о реальном отсутствии мотива ввиду недостаточной изученности некоторых традиций. Последнее обстоятельство не позволяет использовать в качестве функции близости стандартные метрики Евклида или Хэмминга, которые предполагают суммирование совпадений по всем сюжетам, что приведёт к установлению высокой близости между двумя слабо исследованными традициями. В связи с этим были выдвинуты альтернативные функции расстояния между традициями *T[i]* и *T[j]* .

Функция *Sc*(*T[i], T[j])=1- 0.5*{ k*C*(*t[i],t[j] /N +1}*, где *C*(*t[i],t[j])* представляет собой величину статистики критерия Хи-квадрат, при оценивании достоверности связи между двумя дихотомическими разбиениями. *N* - общее количество мотивов в исследуемой базе,

k=1, если мотивы в среднем чаще встречаются в *T[j]* при условии наличия их в *T[i].*

k=-1, если мотивы в среднем реже появляются в *T[j]* при условии наличия их в *T[i].*

Выявление однородных групп традиций. Для выявления групп традиций с близким характером встречаемости мифологических мотивов использовался широко распространённый метод иерархической группировки.

На начальном этапе каждая традиция считалась отдельным кластером.

На каждом шаге происходит объединение кластеров с миниимальным значением усреднённой (по всем парам объектов из разных кластеров) функции расстояния.

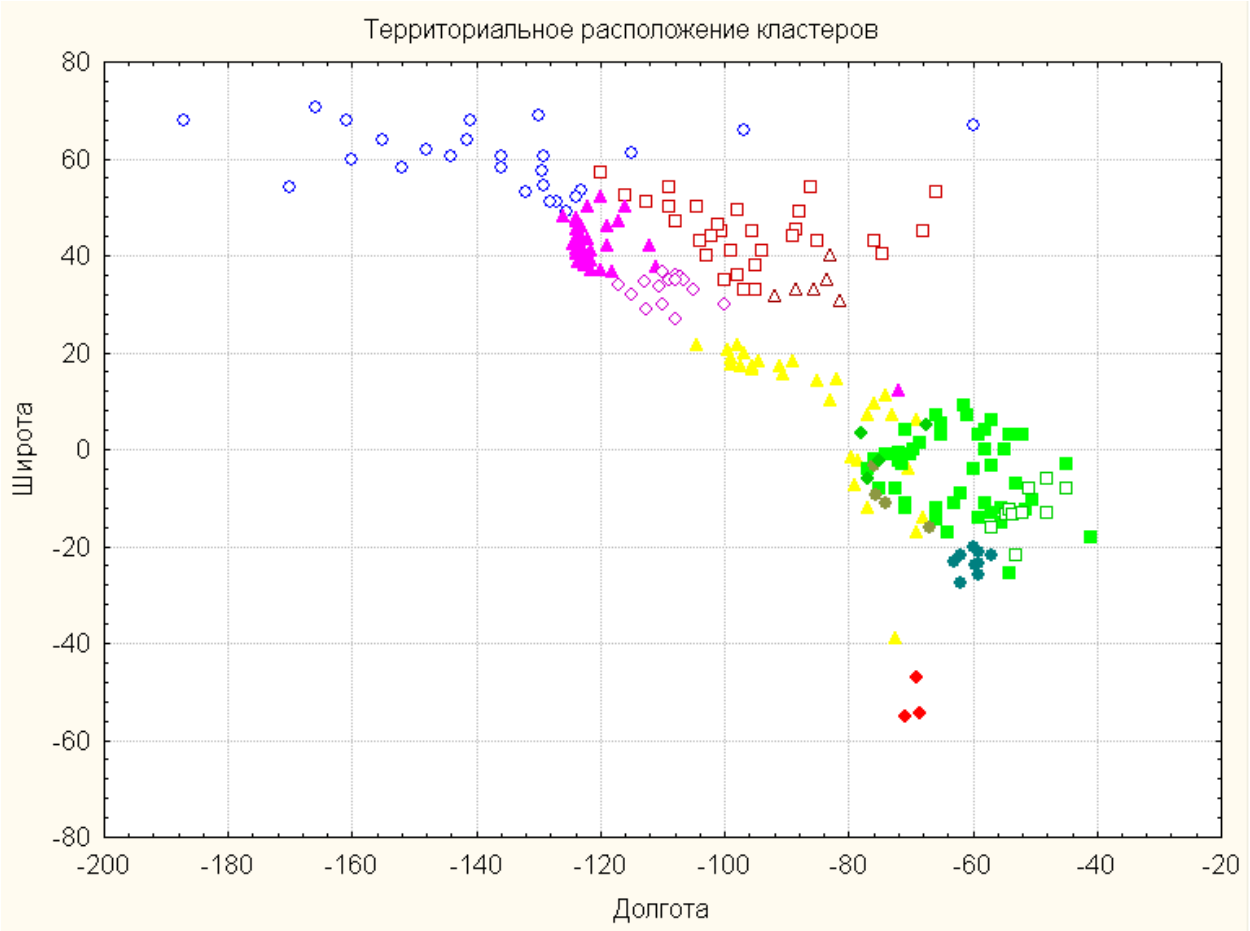
Процесс продолжался до тех пор пока традиции не оказывались объединёнными в заданное исследователем число кластеров.

На первом этапе исследования проводились для индейских традиций Американского континента

Проведённые исследования показали, что традиции в кластерах, полученных согласно сходству мифологических мотивов, оказываются, как правило, также близкими географически.

На приводимых далее рисунках результаты кластеризации показаны в системе географических координат при заданном числе кластеров равным 8 и 11.


Территориальное распределение кластеров
80
60
40
20
0
-20
-40
-60
-80
ШИРОТА
-200
-180
-160
-140
-120
-100
-80
-60
-40
-20
ДОЛГОТА


Территориальное расположение кластеров
80
60
40
20
0
-20
-40
-60
-80
Широта
-200
-180
-160
-140
-120
-100
-80
-60
-40
-20
Долгота

Дополнительным способом оценки сходства между традициями (или группами традиций) *T1* и *T2* является вычисление коэффициента корреляции расстояния до *T1* и *T2* набора других традиций.

Использовался набор всех индейских традиций американского континента.

Таблица 1. *Коэффициенты корреляции между средними расстояниями американских фольклорных традиций до соответствующих пар кластеров.*

	○	□	△	▲	◇	▲	■	○	●	◆	●	◆
○	**1.00**	0.29	0.00	0.33	0.01	-0.42	-0.32	-0.28	-0.17	-0.27	-0.26	0.00
□	0.29	**1.00**	0.48	0.46	0.46	-0.37	-0.58	-0.43	-0.21	-0.46	-0.42	0.01
△	0.00	0.48	**1.00**	0.26	0.47	0.17	-0.30	-0.30	-0.10	-0.13	-0.12	-0.09
▲	0.33	0.46	0.26	**1.00**	0.55	-0.29	-0.37	-0.31	0.08	-0.30	-0.32	0.06
◇	0.02	0.46	0.47	0.55	**1.00**	0.03	-0.41	-0.34	-0.04	-0.27	-0.24	0.10
▲	-0.42	-0.37	0.17	-0.29	0.03	**1.00**	0.38	0.15	0.10	0.45	0.43	0.04
■	-0.32	-0.58	-0.30	-0.37	-0.41	0.38	**1.00**	0.75	0.37	0.64	0.68	0.05
○	-0.28	-0.43	-0.30	-0.31	-0.34	0.15	0.75	**1.00**	0.34	0.38	0.39	-0.01
●	-0.17	-0.21	-0.10	0.08	-0.04	0.10	0.37	0.34	**1.00**	0.17	0.24	0.14
◆	-0.27	-0.46	-0.13	-0.30	-0.27	0.45	0.64	0.38	0.17	**1.00**	0.48	0.05

	-0.26	-0.42	-0.12	-0.32	-0.24	0.43	0.68	0.39	0.24	0.48	1.00	0.08
	0.00	0.01	-0.09	0.06	0.10	0.04	0.05	-0.01	0.14	0.05	0.08	**1.00**

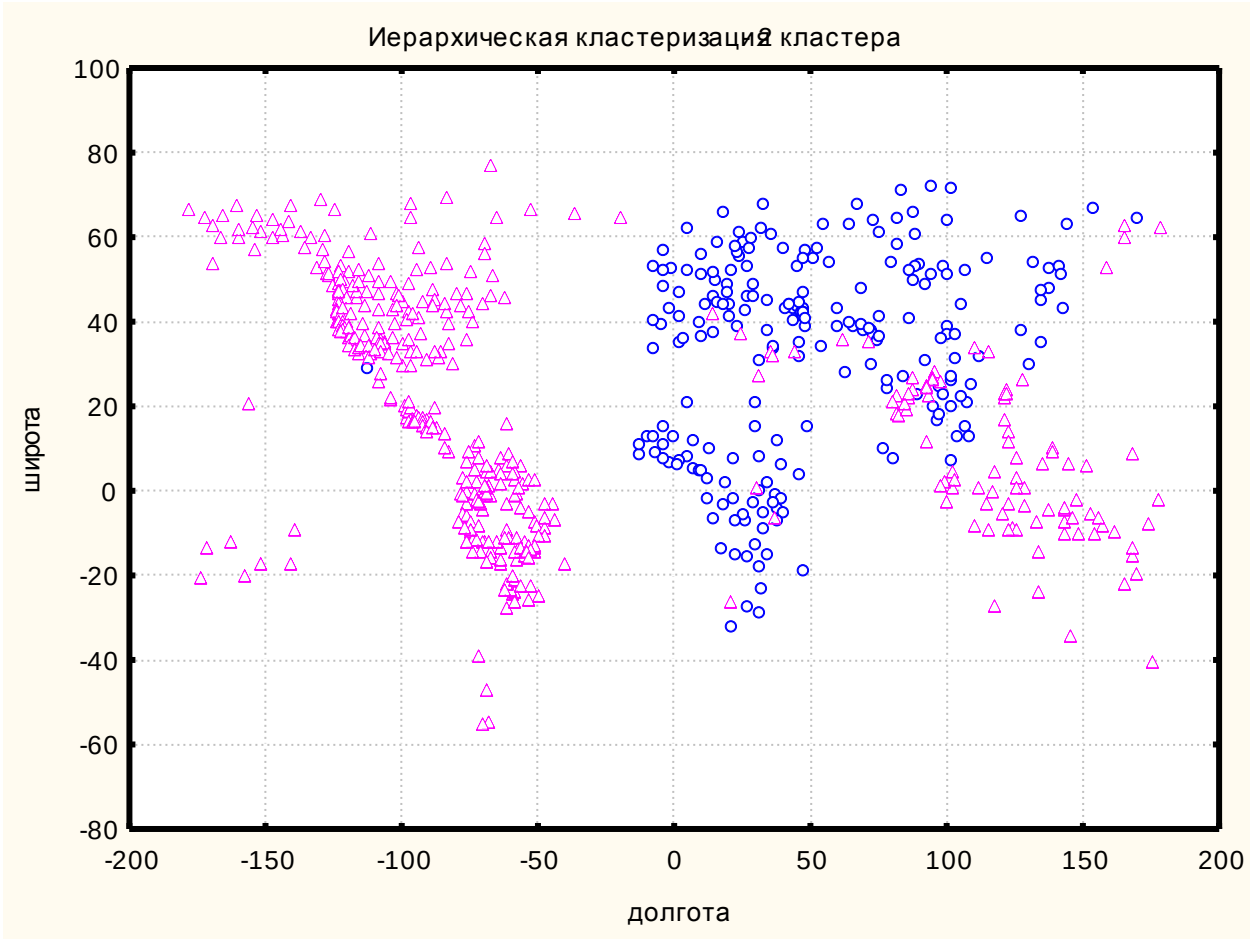
Таблица 2. *Коэффициенты корреляции между средними расстояниями американских Фольклорных традиций для пар (кластер –внеамерикнская традиция).*

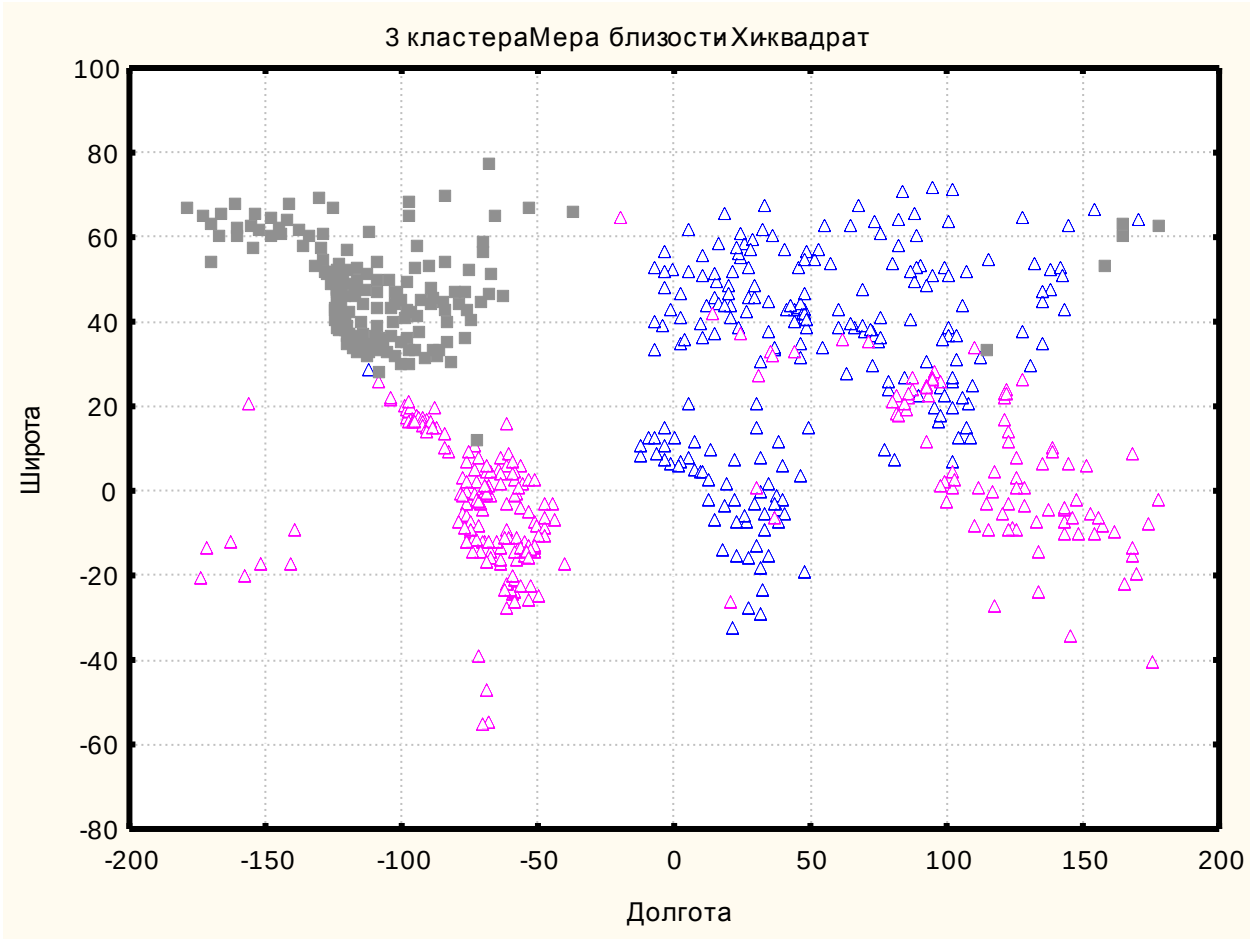
	○	□	△	▲	◇	▲	■	○	●	◆	●	◆
Chinese_	0.13	-0.13	0.22	-0.09	0.16	0.58	0	-0.14	0.01	0.09	0.16	0.04
Garo_Chin_Mizo_Kachari_	0.0025	-0.066	0.33	-0.03	0.15	0.62	0.21	-0.039	0.01	0.26	0.46	0.018
Hadza_Sandawe	-0.42	-0.34	-0.06	-0.16	0.057	0.37	0.47	0.55	0.32	0.46	0.35	0.20
Chukchi	**0.78**	0.38	0.24	0.40	0.31	-0.22	-0.47	-0.49	-0.19	-0.28	-0.32	0.028
Evenk:_Baikal_Amur	0.35	0.54	0.57	0.48	0.61	-0.04	-0.50	-0.46	-0.19	-0.26	-0.27	0.037
Ainu	0.61	0.15	0.11	0.33	0.15	-0.03	-0.14	-0.20	0.16	-0.19	-0.08	0.11
New_Guinea_Papuans	-0.26	-0.49	-0.17	-0.21	-0.24	0.47	0.81	0.60	0.37	0.71	0.52	0.12

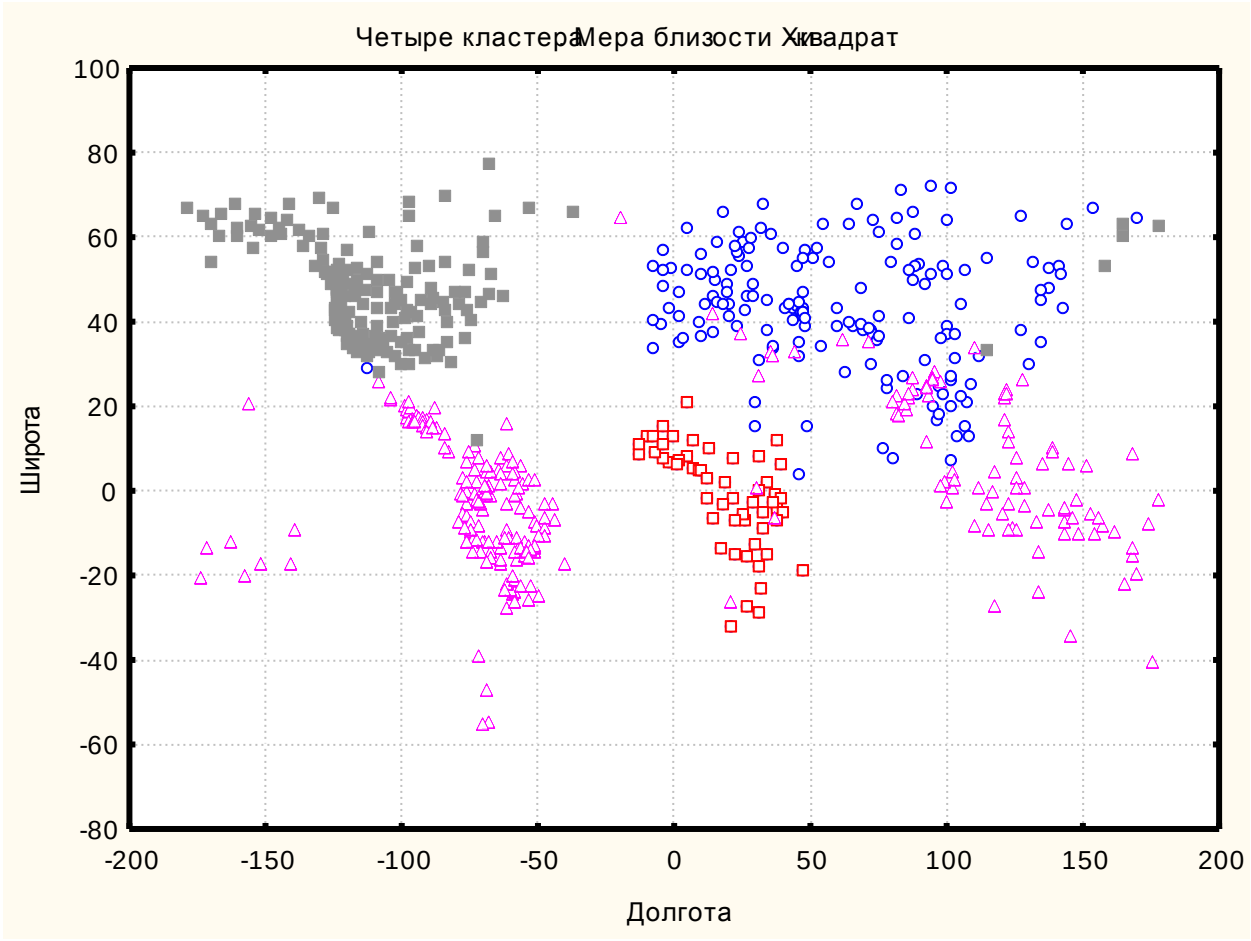
На втором этапе исследования проводились для традиций, распространённых по всему миру

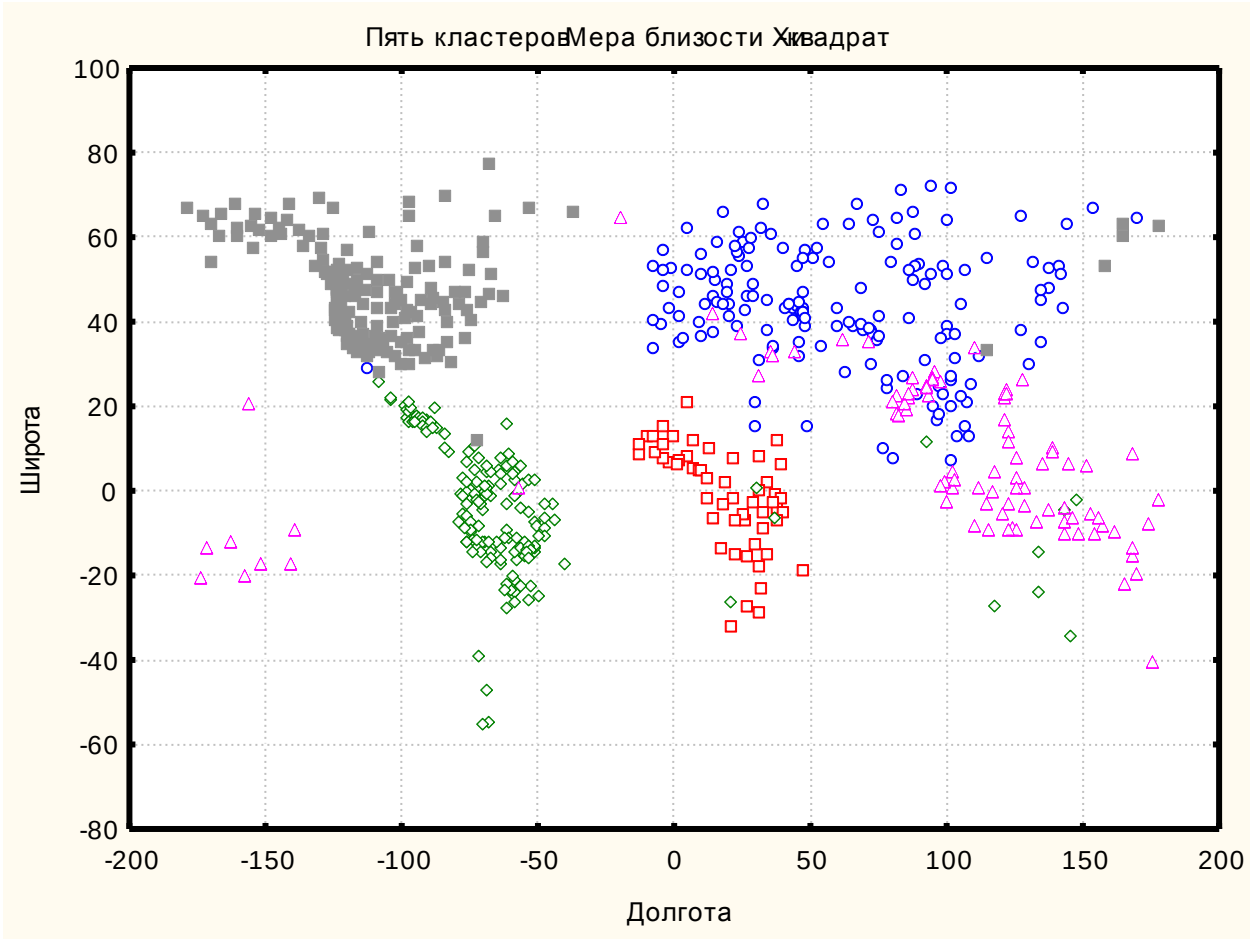
На приводимых далее рисунках результаты кластеризации показаны в системе географических координат при заданном числе кластеров равным от 2 до 8.

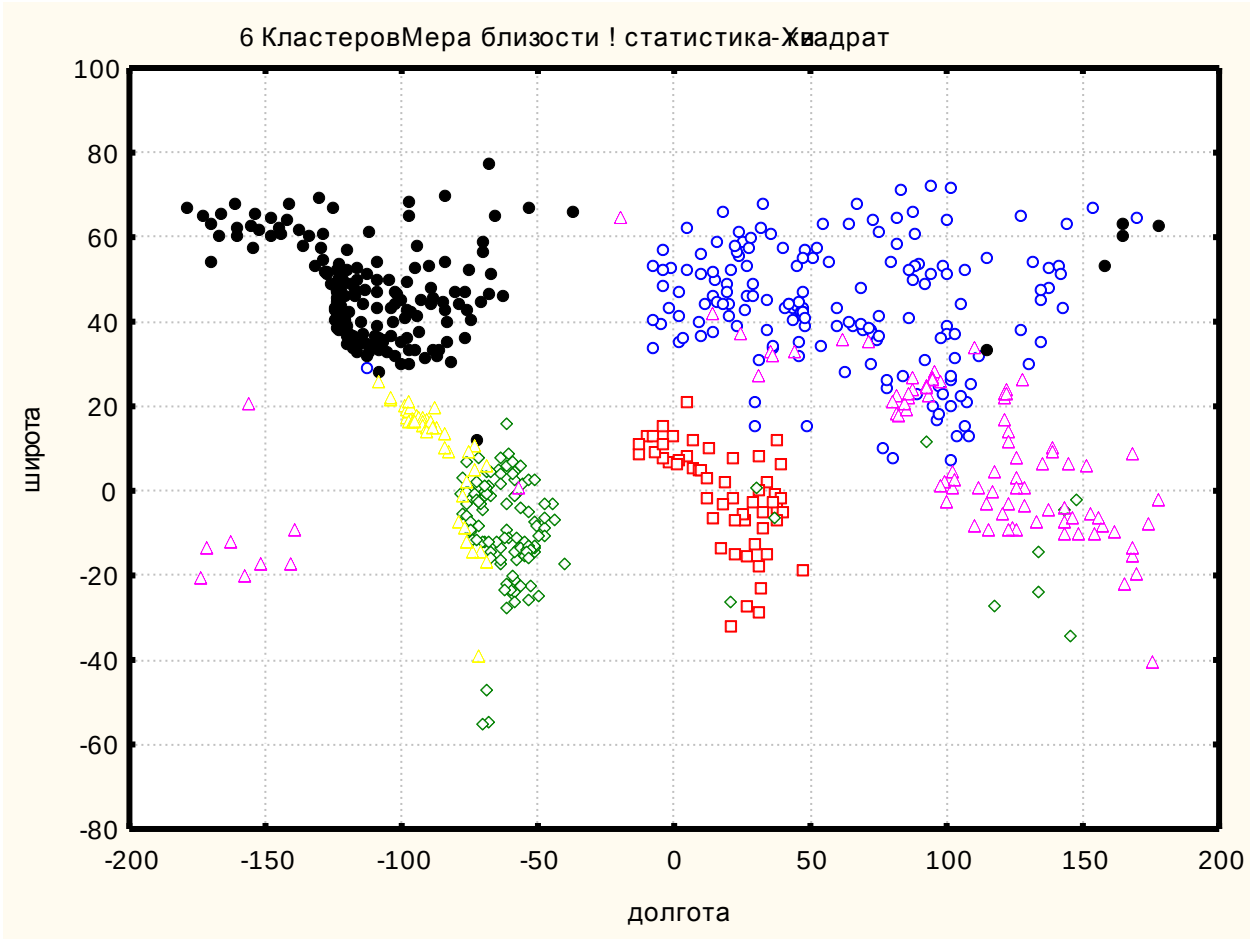
Исследования подтвердили выраженную тенденцию, что традиции в кластерах, полученных согласно сходству мифологических мотивов, оказываются, как правило, также близкими географически.

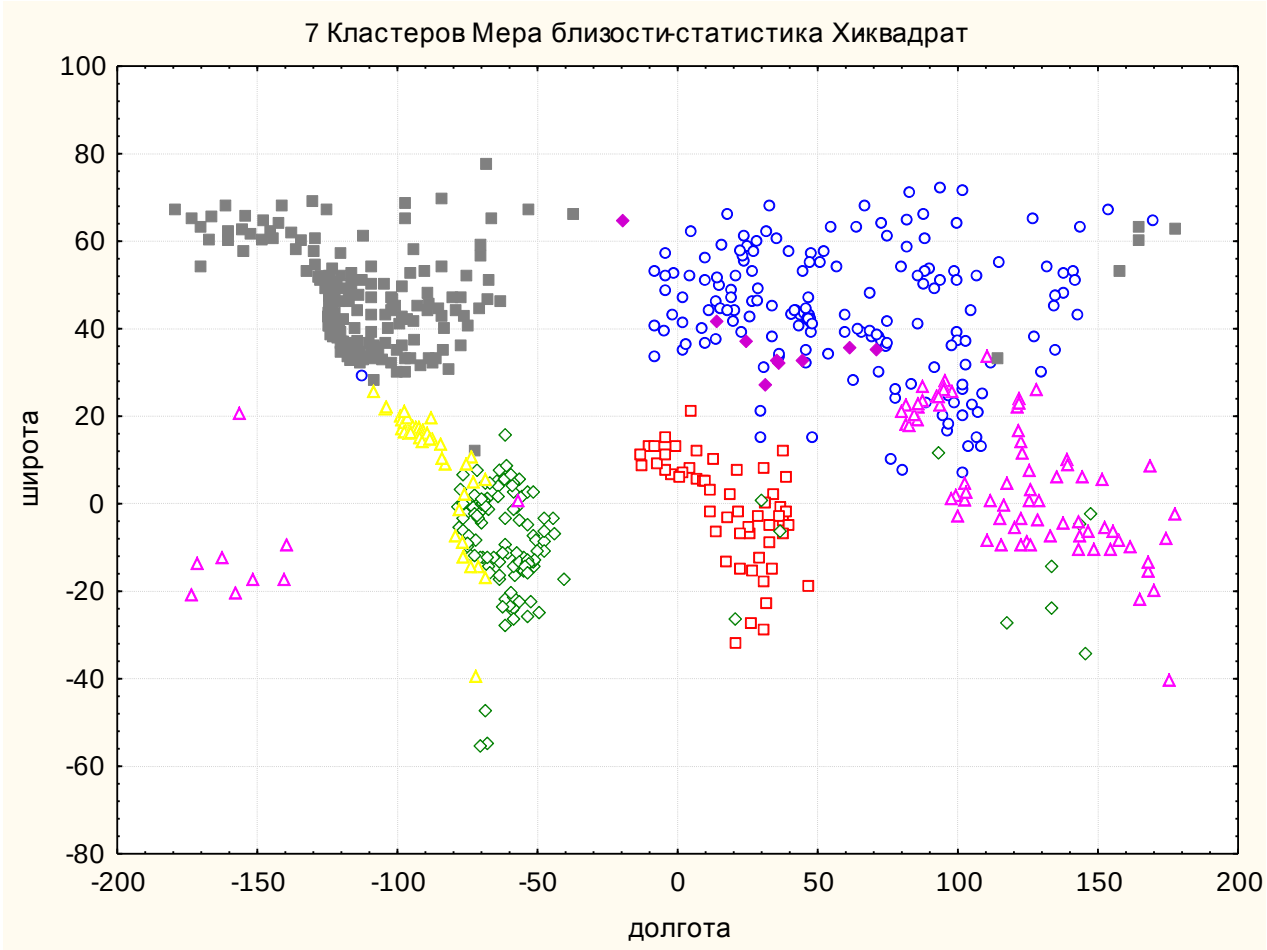

Иерархическая кластеризация 2 кластера
100
80
60
40
20
0
-20
-40
-60
-80
широта
-200
-150
-100
-50
0
50
100
150
200
долгота


3 кластераМера близостиХи-квадрат
100
80
60
40
20
0
-20
-40
-60
-80
Широта
-200
-150
-100
-50
0
50
100
150
200
Долгота


Четыре кластера Мера близости Хи-квадрат
100
80
60
40
20
0
-20
-40
-60
-80
Широта
-200
-150
-100
-50
0
50
100
150
200
Долгота


Пять кластеров Мера близости Хи-квадрат
100
80
60
40
20
0
-20
-40
-60
-80
Широта
-200
-150
-100
-50
0
50
100
150
200
Долгота


6 КластеровМера близости ! статистика-Хи
100
80
60
40
20
0
-20
-40
-60
-80
широта
-200
-150
-100
-50
0
50
100
150
200
долгота



85 0.159000 Ancient_Egypt_____ Afrasian_____ 27.500000 30.700001

99 0.170000 Accad,Assiria,Babylon_____ Afrasian_____ 33.000000 44.000000

101 0.198125 Ugarit,_Phoenicia_____ Afrasian_____ 33.000000 35.000000

140 0.190500 Ancient_Italy_____ Indoeuropean+_____ 42.000000 13.500000

158 0.211875 Ancient_Greece_____ Indoeuropean_____ 37.500000 24.000000

100	0.162375	Old_Testament	Afrasian	32.500000 35.500000
111	0.166125	Zoroastrism,Shah_Name	Indoeuropean	36.000000 61.000000
174	0.124500	Edda	Indoeuropean	65.000000 -20.000000
270	0.092750	Kafirs	Indoeuropean	35.500000