DEEP LEARNING

Неделя 2

Святослав Елизаров, Борис Коваленко, Артем Грачев

28 октября 2017

Высшая школа экономики

ЛИНЕЙНЫЕ МОДЕЛИ

БАЙЕСОВСКАЯ ЛИНЕЙНАЯ РЕГРЕССИЯ

Будем считать, что вектор параметров θ имеет априорное нормальное распределение $N(0, \lambda^{-1}I)$. Выпишем логарифм правдоподобия:

$$\log L(\theta) = \log \prod_{i=1}^{m} N\left(y_i|\theta^T x_i, \sigma^2\right) N\left(\theta|0, \lambda^{-1}I\right) =$$

$$= \sum_{i=1}^{m} -\frac{1}{\sigma^2}(y_n - \theta x_i)^2 - \frac{\lambda}{\sigma^2}\theta^T\theta + const$$

Обратите внимание на дополнительное слагаемое в функционале.

БАЙЕСОВСКАЯ ЛИНЕЙНАЯ РЕГРЕССИЯ

Используя метод апостериорного максимума можно получить замкнутую форму для θ:

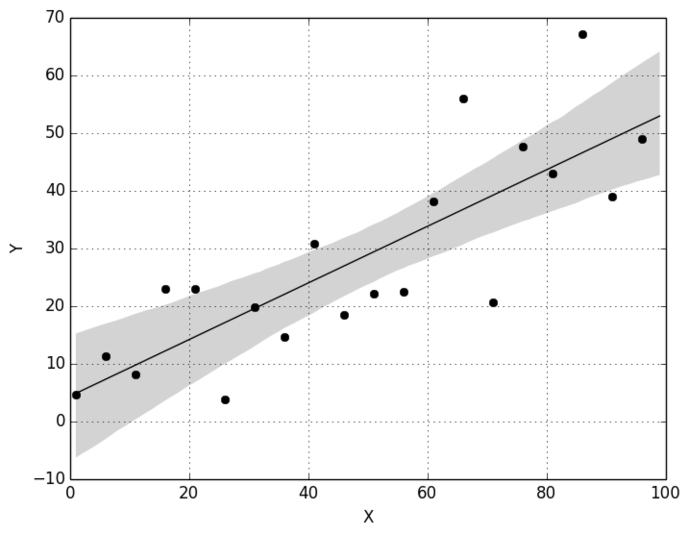
$$\theta = \left(X^T X + \lambda I\right)^{-1} X^T y$$

Используя полученную оценку θ, мы можем найти оценку для таргета y, она будет не точечной оценкой, а распределением:

$$y_i \sim N\left(\frac{1}{\sigma^2} x_i^T A^{-1} X^T y, x_i^T A^T x_i + \sigma^2\right)$$

где $A = \frac{1}{\sigma^2} X^T X + \lambda I$

БАЙЕСОВСКАЯ ЛИНЕЙНАЯ РЕГРЕССИЯ

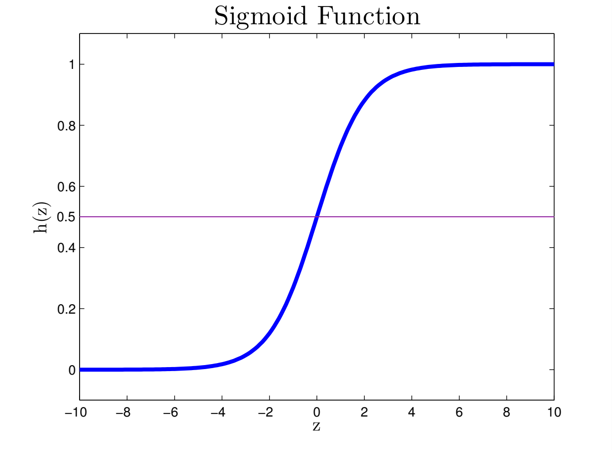


Логистическая регрессия:

$$f(x) = \sigma(\theta_1 x + \theta_0)$$

Где $\sigma(x) = \frac{1}{1+e^{-x}}$

Функция $\sigma(x)$ называется **логистическим сигмоидом** (logistic sigmoid)



Если мы хотим перейти в другое пространство, где выборка разделима линейно, мы можем добавить полиномиальные признаки. Например x^2:

$$f(x) = \sigma(\theta_1 x + \theta_2 x^2 + \theta_0)$$

$$L = -\log\left(\frac{e^{y_{true}}}{\sum\limits_{m} e^{y_m}}\right)$$

Функция softmax переводит оценки в ”вероятности”:

$$softmax(y_i) = \frac{e^{y_i}}{\sum\limits_{m} e^{y_m}}$$

Где y_i – оценка, полученная моделью для i-го класса.

CROSS ENTROPY LOSS

$$L = -\log\left(\frac{e^{y_{true}}}{\sum\limits_m e^{y_m}}\right)$$

В чём смысл этой формулы? Почему она работает?

CROSS ENTROPY LOSS

Количество информации:

$$I(x) = -\log_2 p(x)$$

CROSS ENTROPY LOSS

Количество информации:

$$I(x) = -\log_2 p(x)$$

Информационная энтропия:

$$H(p) = -\sum_x p(x)\log_2 p(x) = EI(x)$$

CROSS ENTROPY LOSS

Количество информации:

$$I(x) = -\log_2 p(x)$$

Информационная энтропия:

$$H(p) = -\sum_x p(x) \log_2 p(x) = EI(x)$$

Кросс-энтропия:

$$H(p, q) = -\sum_x p(x) \log_2 q(x)$$

Кросс-энтропия :

$$H(p,q) = -\sum_x p(x)\log q(x) = -\sum_x p(x)(\log q(x) + \log p(x) - \log p(x)) =$$

$$H(p) + \sum_x p(x)(\log p(x) - \log q(x))$$

Дивергенция Кульбака—Лейблера:

$$D_{KL}(p||q) = \sum_{x} p(x)(\log p(x) - \log q(x))$$

Дивергенция Кульбака—Лейблера:

$$D_{KL}(p||q) = \sum_x p(x)(\log p(x) - \log q(x))$$

- $\forall p, q, D_{KL}(p||q) \geq 0$
- $D_{KL}(p||q) \neq D_{KL}(q||p)$

Дивергенция Кульбака—Лейблера:

$$D_{KL}(p||q) = \sum_x p(x)(\log p(x) - \log q(x))$$

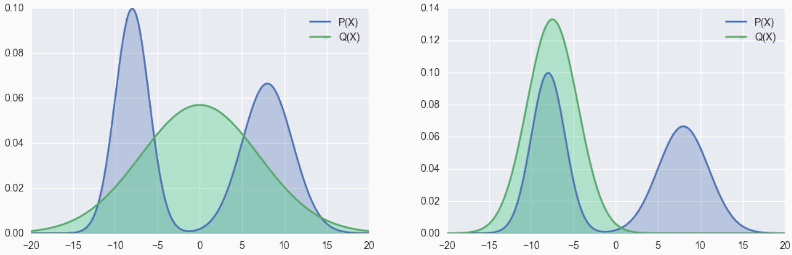
- $\forall p, q, D_{KL}(p||q) \geq 0$
- $D_{KL}(p||q) \neq D_{KL}(q||p)$

Дивергенция Кульбака—Лейблера:

Прямая $D_{KL}(p||q)$ или обратная $D_{KL}(q||p)$?

Дивергенция Кульбака—Лейблера:

Прямая $D_{KL}(p||q)$ или обратная $D_{KL}(q||p)$?



Важной величиной является **взаимная информация** (mutual information).

Рассмотрим две случайные величины x и y, а так же их совместное распределение $p(x, y)$

Важной величиной является **взаимная информация** (mutual information).

Рассмотрим две случайные величины x и y, а так же их совместное распределение $p(x, y)$

Если случайные величины независимы, то

$$p(x, y) = p(x)p(y)$$

$$p(x) = p(x|y)$$

$$p(y) = p(y|x)$$

$$D_{KL}(p(x,y)||p(x)p(y))$$

Называется взаимной информацией и является мерой зависимости двух случайных величин.

Взаимная информация обозначается $I(x;y)$.

CROSS ENTROPY LOSS

Кросс-энтропия:

$$H(p,q) = -\sum_x p(x) \log q(x) = H(p) + D_{KL}(p||q)$$

Данная функция потерь минимизирует KL дивергенцию между истинным и полученным распределением для объектов.

Истинное распределение - $[0, ..., 1, ..., 0]$

Предсказанное распределение - $[0.05, ..., 0.6, ..., 0.1]$

CROSS ENTROPY LOSS

Пример:

Получен вектор оценок для задачи классификации: $[1.2, 5, 1.6]$, истинный класс 1

- Получим вектор ”вероятностей”с помощью софтмакса - $[0.021, 0.947, 0.032]$
- $L = -\log(0.021) = 3.86$

Рассмотрим вероятностную постановку задачи:

$$P(y = 1) = h_\theta(x)$$

$$P(y = 0) = 1 - h_\theta(x)$$

или в компактной форме:

$$P(y|x;\theta) = (h_\theta(x))^y(1 - h_\theta(x))^{(1-y)}$$

Чтобы найти параметры θ, выпишем функцию правдоподобия и будем ее максимизировать

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

$$l(\theta) = \log L(\theta) = \log \prod_{i=1}^{m} P(y|x;\theta) = \log \prod_{i=1}^{m} (h_\theta(x))^y (1 - h_\theta(x))^{(1-y)} =$$

$$\sum_{i=1}^{m} y \log((h_\theta(x))) + (1-y) \log(1 - h_\theta(x))$$

Как оптимизировать данную функцию? Градиентный спуск или матричная форма в замкнутом виде?

Найдем градиент функции для 1 примера:

$$\frac{\partial}{\partial \theta} l(\theta) = \left[y \frac{1}{g(\theta^T x)} - (1-y) \frac{1}{1-\sigma(\theta^T x)} \right] \sigma(\theta^T x)(1-\sigma(\theta^T x))x =$$

$$= (y - h_\theta(x))x$$

правило обновления вектора параметров θ принимает вид:

$$\theta_i = \theta_{i-1} + \alpha(y - h_\theta(x))x$$

На что похожа эта формула?

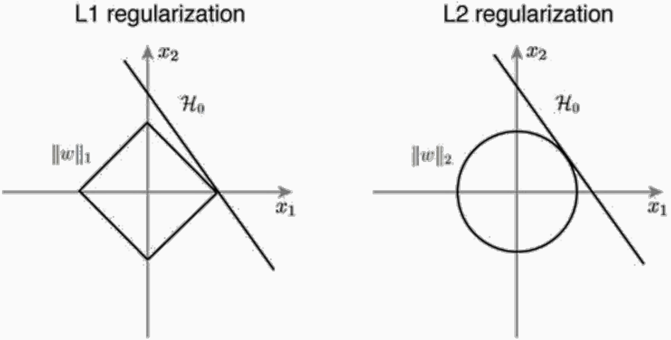
РЕГУЛЯРИЗАЦИЯ

Loss = Data Loss + λ (Regularization Loss)

L2: $R(W) = \sum_i \sum_j W_{ij}^2$

L1: $R(W) = \sum_i \sum_j |W_{ij}|$

Elastic Net: L1 + L2



Обобщенная линейная модель (GLM)

Экспотенциальное семейство распределений:

$$p(y, \eta) = b(y) \exp\left(\eta^T T(y) - a(\eta)\right)$$

η - natural parameter

$T(y)$ - достаточная статистика (sufficient statistic)

$a(\eta)$ - log partition function

Распределения Бернулли и Нормальное распределение $\in$ Экспотенциальное семейство распределений

Целевая переменная имеет Нормальное распределение $\rightarrow$ линейная регрессия

Целевая переменная имеет распределение Бернулли $\rightarrow$ логистическая регрессия

Покажем что Распределения Бернулли и Нормальное действительно $\in$ Экспотенциальное семейство распределений

$$p(y;\phi) = \phi^y(1-\phi)^{1-y} = \exp(y\log(\phi) + (1-y)\log(1-\phi)) =$$

$$= \exp\left(\log[\frac{\phi}{1-\phi}]y + log(1-\phi)\right)$$

$\eta = \log[\frac{\phi}{1-\phi}]$

$b(y) = 1$

$T(y) = y$

$a(\eta) = -\log(1-\phi) = \log\left(1 + \exp(\eta)\right)$

ϕ - ?

GLM

$$p(y;\mu) = \frac{1}{\sqrt{2\pi}\sigma}\,\mathrm{exp}(\frac{1}{2}(y-\mu)^2) = \frac{1}{\sqrt{2\pi}\sigma}\,\mathrm{exp}(\frac{1}{2}y^2)\,\mathrm{exp}(\mu y - \frac{1}{2}\mu^2)$$

$\eta = \mu$

$b(y) = \frac{1}{\sqrt{2\pi}\sigma} exp(\frac{1}{2}y^2)$

$T(y) = y$

$a(\eta) = \frac{1}{2}\mu^2 = \frac{1}{2}\eta^2$

GLM

Функция оценки связи между целевой переменной и признаками:

Для линейной регрессии:

$$h_\theta(x) = E[y|x;\theta] = \mu = \eta = \theta^T x$$

GLM

Функция оценки связи между целевой переменной и признаками:

Для линейной регрессии:

$$h_\theta(x) = E[y|x;\theta] = \mu = \eta = \theta^T x$$

Для логистической регрессии:

$$h_\theta(x) = E[y|x;\theta] = \phi = \frac{1}{1 - \exp^{-\eta}} = \frac{1}{1 - \exp^{-\theta^T x}}$$

GLM

Функция оценки связи между целевой переменной и признаками:

Для линейной регрессии:

$$h_\theta(x) = E[y|x;\theta] = \mu = \eta = \theta^T x$$

Для логистической регрессии:

$$h_\theta(x) = E[y|x;\theta] = \phi = \frac{1}{1 - \exp^{-\eta}} = \frac{1}{1 - \exp^{-\theta^T x}}$$

natural parameter η и признаки x имеют линейную связь $\eta_i = \theta_i^T x$

GLM

Правило обновления признаков для всех моделей этого типа имеет вид:

$$\theta_i = \theta_{i-1} + \alpha(y - h_\theta(x))x$$

Расмотрим еще одну линейную модель - SVM

Основная идея - максимизация зазора между объектами разных классов, оптимизируемый функционал имеет вид:

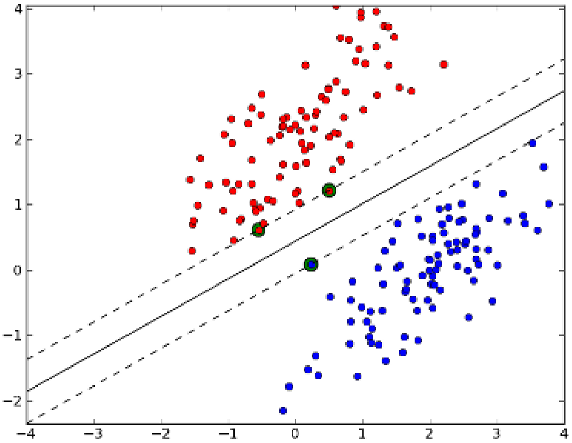
$$J_i(\theta) = \max \sum_{j \neq y_i} \{0, s_j - s_{y_i} + \Delta\}$$

Зависимость между целевой функцией и признаками линейная:

$$s_i^j = \theta_j^T x_i$$

Как оптимизировать такой функционал?

SVM



MULTICLASS SVM (HINGE LOSS)

$$J_i(\theta) = \sum_{j \neq y_i} max(0, \theta_j^T x_i - \theta_{y_i}^T x_i + \Delta)$$

MULTICLASS SVM (HINGE LOSS)

Пример:

Получен вектор скоров для задачи классификации: $[1.2, 5, 1.6]$, параметр $\Delta = 2$, истинный класс 1, loss для данного объекта равен

$$L_i = max(0, 5 - 1.2 + 2) + max(0, 1.6 - 1.2 + 2)$$

L2 Регуляризация

$$R(\theta) = \sum_i \sum_j w_{i,j}^2$$

$$J(\theta) = \frac{1}{N} \sum_i L_i + \lambda R(\theta)$$

ЯДРА

Функция $K(x, y)$ называется ядром, если:

1. $K(x, y) = K(y, x)$
2. $K(x, y)$ неотрицательно определенная

ЯДРА

1. $K(x, y) = \langle x, y \rangle$
2. $K(x, y) = (\langle x, y \rangle + r)^m$
3. $K(x, y) = e^{-\gamma||x-y||^2}$

Опишем в общем виде модель линейной регрессии:

$$f(x) = x'w^T = \langle x', w \rangle$$

где

- $x = (x_1, x_2, \ldots, x_n) \in \Omega$
- $x' = 1 \cup x$, x' – это вектор-строка x, первым (с индексом 0) элементом которой назначена константа 1. В дальнейшем будем подразумевать под x вектор данной конструкции. Иногда эта конструкция называется **bias trick**.
- $w = (w_0, w_1, \ldots, w_n)$, веса модели

Теперь введём следующую модель:

$$f(x) = \psi\left(\sum_{i=0}^{N} w_i \phi_i(x)\right) = \psi\left(\langle w, \phi(x)\rangle\right)$$

Где w_i – веса (параметры модели) при i-м компоненте, w – вектор (матрица или тензор) весов.

Где ϕ_i – базисные функции (налагается требование дифференцируемости). При их помощи можно, например, добавить x^2 как признак.

$\phi(x)$ – вектор (матрица или тензор) значений базисных функций от x.

ψ – функция активации, так же должна быть дифференцируемой. В случае логистической регрессии это сигмоид.

НЕЙРОННЫЕ СЕТИ

Искусственные нейронные сети прямого распространения (feed-forward artificial neural networks):.

- В качестве базисных функций $\phi_i(x)$ возьмём эту же модель.
- Поступим так несколько раз
- Верхним индексом обозначаем уровень вложенности (самый ”глубокий”0)
- Совокупность элементов имеющих один верхний индекс принято называть ”слоем”

Например, так будет выглядеть формула, описывающая простейшую полносвязную сеть с одним скрытым слоем (размерности 3), решающую задачу бинарной классификации для $x \in R^3$:

$$f(x) = \sigma(w_1^2\sigma(w_{11}^1 x_1 + w_{12}^1 x_2 + w_{13}^1 x_3) + \cdots + w_3^2\sigma(w_{31}^1 x_1 + w_{32}^1 x_2 + w_{33}^1 x_3))$$

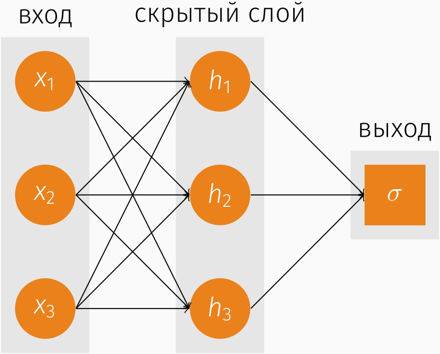
Или в матричном виде. Обратите внимание, что для удобства записи номер слоя теперь обозначен в нижнем индексе:

$$f(x) = \sigma(W_2^T\sigma(W_1^T x))$$

ГРАФ НЕЙРОННОЙ СЕТИ

Теперь, если мы представим каждое слагаемое как вершину в графе и проведём ребра между теми вершинами которые представлены в одной сумме с ненулевыми весами, то получим привычный граф искусственной нейронной сети.

ГРАФ НЕЙРОННОЙ СЕТИ



Где $h_i = \sigma(w^1_{i1}x_1 + w^1_{i2}x_2 + w^1_{i3}x_3)$

В качестве функций активации обычно используют:

- Уже хорошо знакомый нам $\sigma(x)$
- Гиперболический тангенс $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}} = 2\sigma(2x) - 1$
- Линейный выпрямитель $ReLU(x) = \max(0, x)$
- $LReLU(x) = \max(x, \alpha x)$, где $\alpha \leq 1$

УНИВЕРСАЛЬНЫЙ АППРОКСИМАТОР

- **Булевы функции.** Любая булева функция может быть представлена в точности какой-либо двухслойной нейронной сетью. При этом число нейроннов в худшем случае может быть экспоненциально большим от числа входов.
- **Непрерывные функции.** Любая ограниченная функция может быть приближена с произвольной точностью (в конечной норме) с помощью нейронной сети с двумя слоями нейроннов. (Cybenko 1989; Hornik et al. 1989). Теорема сформулирована для сигмодиной функции активации. Количество нейроноов зависит от аппрокисимируемой функции.
- **Произовольная функция.** Любая функция может быть приближена с произвольной точностью с помощью нейронной сетью с тремя слоями нейронов. (Cybenko 1989)

Mitchell T. Machine Learning. McGraw-Hill Science/Engineering/Math, 1997. ISBN 0-07-042807-7

НЕМНОГО ИСТОРИИ

- 1943 – первая работа о том, как нейроны могут работать
- 1949 – модель Хебба
- 1953 – перцептрон Розенблатта
- 1974, 1986 — алгоритм обратного распространения ошибки (backpropagation) Галушкин, Вербос
- 1997 — RNN и LSTM
- 2006 – машины Больцмана
- 2012 — победа AlexNet на ImageNet-2012
- 2013 — VAE, Word2Vec
- 2014 — Generative Adversarial Networks
- 2015 — DQN и Atari games
- 2016 — AlphaGo