Мат.модели машинного обучения: линейные модели и метод стохастического градиента

Воронцов Константин Вячеславович
k.v.vorontsov@phystech.edu
http://www.MachineLearning.ru/wiki?title=User:Vokov

Этот курс доступен на странице вики-ресурса
http://www.MachineLearning.ru/wiki
«Машинное обучение (курс лекций, К.В.Воронцов)»

МФТИ • 13 сентября 2024

Содержание

Задача обучения регрессии — это оптимизация

Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell$, $x_i \in \mathbb{R}^n$, $y_i = y(x_i) \in \mathbb{R}$

1. Модель регрессии — *линейная* с параметром $w \in \mathbb{R}^n$:
$$a(x, w) = \langle x, w \rangle = \sum_{j=1}^n w_j f_j(x)$$
2. Функция потерь — *квадратичная*:
$$\mathscr{L}(w, x) = \bigl(a(x, w) - y(x)\bigr)^2$$
3. Метод обучения — *метод наименьших квадратов*:
$$Q(w) = \sum_{i=1}^\ell \mathscr{L}(w, x_i) = \sum_{i=1}^\ell \bigl(a(x_i, w) - y_i\bigr)^2 \to \min_w$$
4. Проверка по тестовой выборке $X^k = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$:
$$\tilde{Q}(w) = \frac{1}{k} \sum_{i=1}^k \bigl(a(\tilde{x}_i, w) - \tilde{y}_i\bigr)^2$$

Задача обучения классификации — тоже оптимизация

Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell, \;\; x_i \in \mathbb{R}^n, \;\; y_i \in \{-1, +1\}$

1. Модель классификации — *линейная* с параметром $w \in \mathbb{R}^n$:
$$a(x, w) = \operatorname{sign}\langle x, w\rangle = \operatorname{sign} \sum_{j=1}^n w_j f_j(x)$$
2. Функция потерь — бинарная или её аппроксимация:
$$\mathscr{L}(w, x) = \bigl[a(x, w) y(x) < 0\bigr] = \bigl[\langle x, w\rangle y(x) < 0\bigr] \leqslant L\bigl(\langle x, w\rangle y(x)\bigr)$$
3. Метод обучения — *минимизация эмпирического риска*:
$$Q(w) = \sum_{i=1}^\ell \mathscr{L}(w, x_i) = \sum_{i=1}^\ell \bigl[\langle x_i, w\rangle y_i < 0\bigr] \leqslant \sum_{i=1}^\ell L\bigl(\langle x_i, w\rangle y_i\bigr) \to \min_w$$
4. Проверка по тестовой выборке $X^k = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$:
$$\tilde{Q}(w) = \frac{1}{k} \sum_{i=1}^k \bigl[\langle \tilde{x}_i, w\rangle \tilde{y}_i < 0\bigr]$$

Задача многоклассовой классификации (multiclass classification)

Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell$, $x_i \in \mathbb{R}^n$, $y_i = y(x_i) \in Y$

1. Модель классификации — *линейная*, $w = \bigl(w_y : y \in Y\bigr)$:
$$a(x, w) = \arg\max_{y \in Y} \langle x, w_y \rangle$$
2. Функция потерь — бинарная или её аппроксимация:
$$\mathscr{L}(w, x) = \sum_{z \neq y(x)} \bigl[\langle x, w_{y(x)} \rangle < \langle x, w_z \rangle\bigr] \leqslant \sum_{z \neq y(x)} L\bigl(\langle x, w_{y(x)} - w_z \rangle\bigr)$$
3. Метод обучения — *минимизация эмпирического риска*:
$$Q(w) = \sum_{i=1}^{\ell} \sum_{z \neq y_i} L\bigl(\langle x_i, w_{y_i} - w_z \rangle\bigr) \to \min_w$$
4. Проверка по тестовой выборке $X^k = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$

Разделяющие классификаторы (margin-based classifier)

Бинарный классификатор: $a(x, w) = \operatorname{sign} g(x, w)$, $\quad Y = \{-1, +1\}$

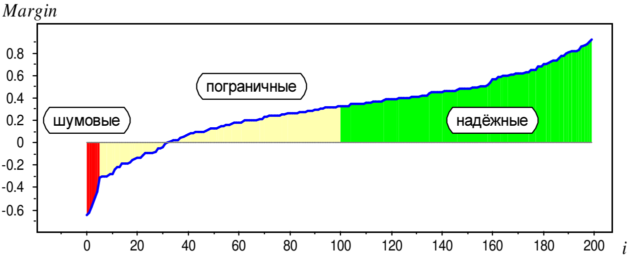
$g(x, w)$ — *разделяющая (дискриминантная) функция*

$x\colon g(x, w) = 0$ — разделяющая поверхность между классами

$M_i(w) = g(x_i, w) y_i$ — *отступ* (margin) объекта x_i

$M_i(w) < 0 \iff$ алгоритм $a(x, w)$ ошибается на x_i

Ранжирование объектов по возрастанию отступов $M_i(w)$:



Разделяющие классификаторы (margin-based classifier)

Многоклассовый классификатор: $a(x, w) = \arg\max\limits_{y \in Y} g_y(x, w_y)$

$g_y(x, w_y)$ — *дискриминантная функция* класса $y \in Y$

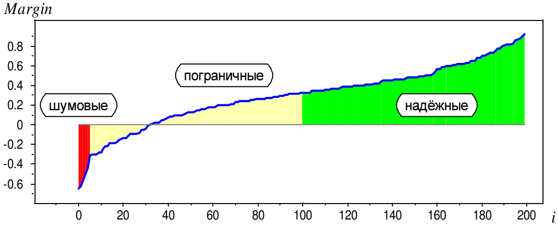
$x\colon g_y(x, w_y) = g_z(x, w_z)$ — разделяющая поверхность между y, z

$M_{iy}(w) = g_{y_i}(x_i, w_{y_i}) - g_y(x_i, w_y)$ — *отступ* объекта x_i от класса y

$M_i(w) = \min\limits_{y \neq y_i} M_{iy}(w)$ — *отступ* (margin) объекта x_i

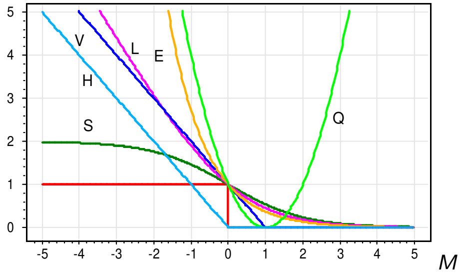
$M_i(w) < 0 \iff$ алгоритм $a(x, w)$ ошибается на x_i

Ранжирование объектов по возрастанию отступов $M_i(w)$:



Непрерывные аппроксимации пороговой функции потерь

Часто используемые непрерывные функции потерь $L(M)$:



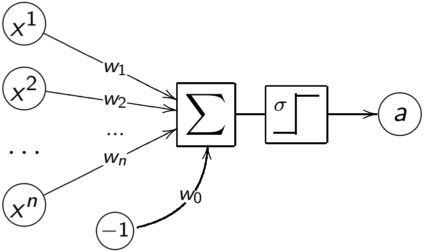
$V(M) = (1 - M)_+$ — кусочно-линейная (SVM);
$H(M) = (-M)_+$ — кусочно-линейная (Hebb's rule);
$L(M) = \log_2(1 + e^{-M})$ — логарифмическая (LR);
$Q(M) = (1 - M)^2$ — квадратичная (FLD);
$S(M) = 2(1 + e^M)^{-1}$ — сигмоидная (ANN);
$E(M) = e^{-M}$ — экспоненциальная (AdaBoost);
$[M < 0]$ — пороговая функция потерь.

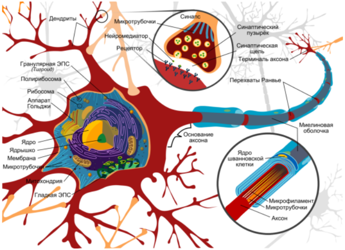
Линейный классификатор — математическая модель нейрона

Линейная модель нейрона МакКаллока-Питтса [1943]:

$$a(x, w) = \sigma(\langle w, x\rangle) = \sigma\left(\sum_{j=1}^{n} w_j f_j(x) - w_0\right),$$

$\sigma(z)$ — функция активации (например, sign),
w_j — весовые коэффициенты синаптических связей,
w_0 — порог активации,
$w, x \in \mathbb{R}^{n+1}$, если ввести константный признак $f_0(x) \equiv -1$

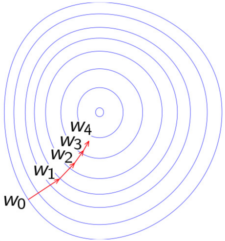




Градиентный метод численной минимизации

Минимизация эмпирического риска:

$$Q(w) = \sum_{i=1}^{\ell} \mathscr{L}(w, x_i) \to \min_w$$



Метод *градиентного спуска*:

$$w^{(0)} := \text{начальное приближение;}$$

$$w^{(t+1)} := w^{(t)} - h \cdot \nabla Q(w^{(t)}), \qquad \nabla Q(w) = \left(\frac{\partial Q(w)}{\partial w_j}\right)_{j=0}^{n},$$

где h — *градиентный шаг*, называемый также *темпом обучения*.

$$w^{(t+1)} := w^{(t)} - h \sum_{i=1}^{\ell} \nabla \mathscr{L}\big(w^{(t)}, x_i\big)$$

Идея ускорения сходимости:

брать объекты x_i по одному и сразу обновлять вектор весов

Алгоритм SG (Stochastic Gradient)

Вход: выборка X^ℓ, темп обучения h, темп забывания λ;
Выход: вектор весов w;

инициализировать веса w_j, $j = 0, \dots, n$;
инициализировать оценку функционала:
  $Q :=$ среднее $\mathscr{L}(w, x_i)$ по случайному подмножеству $\{x_i\}$;
повторять
  выбрать объект x_i из X^ℓ случайным образом;
  вычислить потерю: $\varepsilon_i := \mathscr{L}(w, x_i)$;
  сделать градиентный шаг: $w := w - h\nabla\mathscr{L}(w, x_i)$;
  оценить функционал: $Q := \lambda\varepsilon_i + (1 - \lambda)Q$;
пока значение Q и/или веса w не сойдутся;

Robbins, H., Monro S. A stochastic approximation method // Annals of Mathematical Statistics, 1951, 22 (3), p. 400–407.

Откуда взялась рекуррентная оценка функционала?

Проблема: вычисление оценки Q по всей выборке $x_1, \ldots, x_\ell$ намного дольше градиентного шага по одному объекту x_i.

Решение: использовать приближённую рекуррентную формулу.

Среднее арифметическое:

$$\bar{Q}_m = \tfrac{1}{m}\varepsilon_m + \tfrac{1}{m}\varepsilon_{m-1} + \tfrac{1}{m}\varepsilon_{m-2} + \ldots$$

$$\bar{Q}_m = \tfrac{1}{m}\varepsilon_m + (1 - \tfrac{1}{m})\bar{Q}_{m-1}$$

Экспоненциальное скользящее среднее:

$$\bar{Q}_m = \lambda\varepsilon_m + (1-\lambda)\lambda\varepsilon_{m-1} + (1-\lambda)^2\lambda\varepsilon_{m-2} + \ldots$$

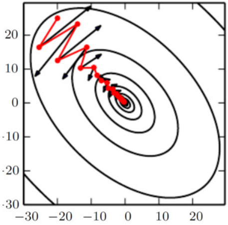
$$\bar{Q}_m = \lambda\varepsilon_m + (1-\lambda)\bar{Q}_{m-1}$$

Параметр λ (порядка $\frac{1}{m}$) — *темп забывания* предыстории ряда.

Метод накопления инерции (momentum)

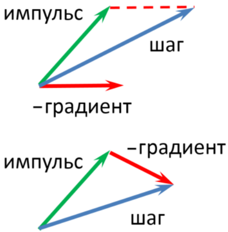
Momentum — экспоненциальное скользящее среднее градиента по последним $\approx \frac{1}{1-\gamma}$ итерациям [Б.Т.Поляк, 1964]:

$$v := \gamma v + (1-\gamma)\mathscr{L}'(w, x_i)$$
$$w := w - hv$$



NAG (Nesterov's accelerated gradient) — стохастический градиент с инерцией [Ю.Е.Нестеров, 1983]:

$$v := \gamma v + (1-\gamma)\mathscr{L}'(w - h\gamma v, x_i)$$
$$w := w - hv$$



Варианты инициализации весов

1. $w_j := 0$ для всех $j = 0, \ldots, n$;
2. небольшие случайные значения:
   $w_j := \text{random}\left(-\frac{1}{2n}, \frac{1}{2n}\right)$;
3. $w_j := \frac{\langle y, f_j \rangle}{\langle f_j, f_j \rangle}$, $f_j = \bigl(f_j(x_i)\bigr)_{i=1}^{\ell}$ — вектор значений признака.

   Эта оценка w оптимальна, если
   1) функция потерь квадратична и
   2) признаки некоррелированы, $\langle f_j, f_k \rangle = 0$, $j \neq k$.
4. обучение по небольшой случайной подвыборке объектов;
5. мультистарт: многократные запуски из разных случайных начальных приближений и выбор лучшего решения.

Варианты порядка предъявления объектов

Возможны варианты:

1. *перетасовка объектов* (shuffling): попеременно брать объекты из разных классов;
2. чаще брать объекты, на которых ошибка больше: чем меньше M_i, тем больше вероятность взять объект;
3. чаще брать объекты, на которых уверенность меньше: чем меньше $|M_i|$, тем больше вероятность взять объект;
4. вообще не брать «хорошие» объекты, у которых $M_i > \mu_+$ (при этом немного ускоряется сходимость);
5. вообще не брать объекты-«выбросы», у которых $M_i < \mu_-$ (при этом может улучшиться качество классификации);

Параметры μ_+, μ_- придётся подбирать.

Варианты выбора градиентного шага

1. сходимость гарантируется (для выпуклых функций) при

$$h_t \to 0, \quad \sum_{t=1}^{\infty} h_t = \infty, \quad \sum_{t=1}^{\infty} h_t^2 < \infty,$$

в частности можно положить $h_t = 1/t$;

2. *метод скорейшего градиентного спуска* основан на поиске оптимального *адаптивного шага* h^*;

$$\mathscr{L}\bigl(w - h\nabla\mathscr{L}(w, x_i), x_i\bigr) \to \min_h,$$

При квадратичной функции потерь $h^* = \|x_i\|^{-2}$.

3. пробные случайные шаги для «выбивания» итерационного процесса из локальных минимумов;
4. метод Левенберга-Марквардта (второго порядка)

Диагональный метод Левенберга-Марквардта

Метод Ньютона-Рафсона, $\mathscr{L}(w, x_i) \equiv L(\langle w, x_i\rangle y_i)$:

$$w := w - h\bigl(\mathscr{L}''(w, x_i)\bigr)^{-1}\nabla\mathscr{L}(w, x_i),$$

где $\mathscr{L}''(w, x_i) = \Bigl(\frac{\partial^2\mathscr{L}(w,x_i)}{\partial w_j \partial w_{j'}}\Bigr)$ — гессиан, $n{\times}n$-матрица

Эвристика. Считаем, что гессиан диагонален:

$$w_j := w_j - h\left(\frac{\partial^2\mathscr{L}(w, x_i)}{\partial w_j^2} + \mu\right)^{-1}\frac{\partial\mathscr{L}(w, x_i)}{\partial w_j},$$

h — темп обучения, можно полагать $h = 1$
μ — параметр, предотвращающий обнуление знаменателя.

Отношение h/μ есть темп обучения на ровных участках функционала $\mathscr{L}(w, x_i)$, где вторая производная обнуляется.

Проблема переобучения

Возможные причины переобучения:

- слишком мало объектов, слишком много признаков
- линейная зависимость (мультиколлинеарность) признаков:

  пусть построен классификатор: $a(x, w) = \operatorname{sign}\langle w, x\rangle$
  мультиколлинеарность: $\exists u \in \mathbb{R}^n\colon\ \forall x \in X\ \ \langle u, x\rangle = 0$
  неединственность решения: $\forall \gamma \in \mathbb{R}\ \ a(x, w) = \operatorname{sign}\langle w{+}\gamma u, x\rangle$

Проявления переобучения:

- слишком большие веса $|w_j|$ разных знаков
- неустойчивость дискриминантной функции $\langle w, x\rangle$
- $Q(X^\ell) \ll Q(X^k)$

Простой способ уменьшить переобучение:

- регуляризация $\|w\| \to \min$ (сокращение весов, weight decay)

Регуляризация (сокращение весов, weight decay)

Штраф за увеличение нормы вектора весов:

$$\widetilde{\mathscr{L}}(w, x_i) = \mathscr{L}(w, x_i) + \frac{\tau}{2}\|w\|^2 = \mathscr{L}(w, x_i) + \frac{\tau}{2}\sum_{j=1}^{n} w_j^2 \to \min_w .$$

Градиент:

$$\nabla\widetilde{\mathscr{L}}(w, x_i) = \nabla\mathscr{L}(w, x_i) + \tau w.$$

Модификация градиентного шага:

$$w := w(1 - h\tau) - h\nabla\mathscr{L}(w, x_i).$$

Методы подбора коэффициента регуляризации τ:

1. скользящий контроль
2. стохастическая адаптация

SG: Достоинства и недостатки

Достоинства:

1. легко реализуется
2. легко обобщается на любые $g(x, w)$, $\mathscr{L}(w, x)$
3. легко добавить регуляризацию
4. возможно динамическое (потоковое) обучение
5. на сверхбольших выборках можно получить неплохое решение, даже не обработав все (x_i, y_i)
6. подходит для задач с большими данными

Недостатки:

1. подбор комплекса эвристик является искусством (не забыть про переобучение, застревание, расходимость)

Принцип максимума правдоподобия

Пусть $X \times Y$ — в.п. с плотностью $p(x, y)$
Пусть X^ℓ — простая (i.i.d.) выборка: $(x_i, y_i)_{i=1}^\ell \sim p(x, y)$
Задача: по выборке X^ℓ оценить плотность $p(x, y)$

$p(x, y) = P(y|x, w)p(x)$ — параметризация плотности:
$P(y|x, w)$ — модель условной вероятности класса y
$p(x)$ — неизвестное и непараметризуемое распределение на X

Максимум правдоподобия (Maximum Likelihood Estimate, MLE):

$$p(X^\ell, w) = \prod_{i=1}^{\ell} p(x_i, y_i) = \prod_{i=1}^{\ell} P(y_i|x_i, w)\cancel{p(x_i)} \to \max_w$$

Максимум логарифма правдоподобия (log-likelihood, log-loss):

$$Q_{\text{MLE}}(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i, w) \to \max_w$$

Связь правдоподобия и аппроксимации эмпирического риска

Максимизация правдоподобия в задаче классификации, где $P(y|x,w)$ — модель *условной вероятности класса* y:

$$Q_{\text{MLE}}(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i,w) \to \max_w$$

Минимизация аппроксимированного эмпирического риска, где $g(x,w)$ — модель *разделяющей поверхности*, $Y = \{\pm 1\}$:

$$Q_{\text{ERM}}(w) = \sum_{i=1}^{\ell} L\bigl(y_i g(x_i,w)\bigr) \to \min_w;$$

Эти два принципа эквивалентны, если положить

$$-\log P(y_i|x_i,w) = L\bigl(y_i g(x_i,w)\bigr).$$

$$\boxed{\text{модель } P(y|x,w)} \rightleftarrows \boxed{\text{модель } g(x,w) \text{ и } L(M)}.$$

Вероятностный смысл регуляризации

$P(y|x,w)$ — вероятностная модель данных;
$p(w;\gamma)$ — априорное распределение параметров модели;
γ — вектор *гиперпараметров*;

Теперь не только появление выборки X^ℓ,
но и появление модели w также полагается стохастическим.

Совместное правдоподобие данных и модели:

$$p(X^\ell, w) = p(X^\ell|w)\, p(w;\gamma).$$

Принцип максимума апостериорной вероятности
(Maximum a Posteriori Probability, MAP):

$$Q_{\text{MAP}}(w) = \ln p(X^\ell, w) = \underbrace{\sum_{i=1}^{\ell} \log P(y_i|x_i,w)}_{Q_{\text{MLE}}(w)} + \underbrace{\log p(w;\gamma)}_{\text{регуляризатор}} \to \max_w$$

Примеры: априорные распределения Гаусса и Лапласа

Пусть веса w_j независимы, $\mathsf{E}w_j = 0$, $\mathsf{D}w_j = C$.

Распределение Гаусса и квадратичный (L_2) регуляризатор:

$$p(w; C) = \frac{1}{(2\pi C)^{n/2}} \exp\left(-\frac{\|w\|^2}{2C}\right), \quad \|w\|^2 = \sum_{j=1}^{n} w_j^2,$$

$$-\ln p(w; C) = \tfrac{1}{2C}\|w\|^2 + \cancel{\mathrm{const}}$$

Распределение Лапласа и абсолютный (L_1) регуляризатор:

$$p(w; C) = \frac{1}{(2C)^n} \exp\left(-\frac{\|w\|}{C}\right), \quad \|w\| = \sum_{j=1}^{n} |w_j|,$$

$$-\ln p(w; C) = \tfrac{1}{C}\|w\| + \cancel{\mathrm{const}}$$

C — гиперпараметр, $\tau = \frac{1}{C}$ — коэффициент регуляризации.

Двухклассовая логистическая регрессия

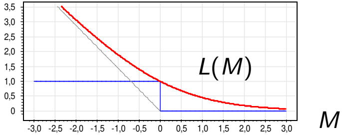
Линейная модель классификации для двух классов $Y = \{-1, 1\}$:

$$a(x, w) = \operatorname{sign}\langle w, x\rangle, \quad x, w \in \mathbb{R}^n.$$

Отступ $M = \langle w, x\rangle y$.

Логарифмическая функция потерь:

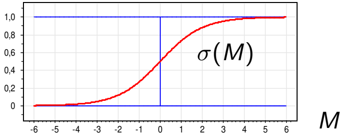
$$L(M) = \log\bigl(1 + e^{-M}\bigr).$$



Модель условной вероятности:

$$P(y|x, w) = \sigma(M) = \frac{1}{1+e^{-M}},$$

где $\sigma(M)$ — сигмоидная функция,
важное свойство: $\sigma(M) + \sigma(-M) = 1$.



Максимизация правдоподобия (logistic loss) с регуляризацией:

$$Q_{\text{MAP}}(w) = \sum_{i=1}^{\ell} \log\bigl(1 + \exp(-\langle w, x_i\rangle y_i)\bigr) + \frac{\tau}{2}\|w\|^2 \;\to\; \min_w$$

Многоклассовая логистическая регрессия

Линейный классификатор при произвольном числе классов $|Y|$:

$$a(x) = \arg\max_{y \in Y} \langle w_y, x \rangle, \quad x, w_y \in \mathbb{R}^n.$$

Вероятность того, что объект x относится к классу y:

$$P(y|x, w) = \frac{\exp\langle w_y, x \rangle}{\sum\limits_{z \in Y} \exp\langle w_z, x \rangle} = \mathop{\mathrm{SoftMax}}_{y \in Y} \langle w_y, x \rangle,$$

функция SoftMax: $\mathbb{R}^Y \to \mathbb{R}^Y$ переводит произвольный вектор в нормированный вектор дискретного распределения.

Максимизация правдоподобия (log-loss) с регуляризацией:

$$Q_{\text{MAP}}(w) = \sum_{i=1}^{\ell} \log P(y_i|x_i, w) - \frac{\tau}{2} \sum_{y \in Y} \|w_y\|^2 \;\to\; \max_{w}.$$

Пример. Бинаризация признаков и скоринговая карта

Задача кредитного скоринга:

- x_i — заёмщики
- $y_i = -1\,(\text{bad}), +1\,(\text{good})$

Бинаризация признаков $f_j(x)$:

$$b_{jk}(x) = \bigl[f_j(x) \text{ из } k\text{-го интервала}\bigr]$$

Линейная модель классификации:

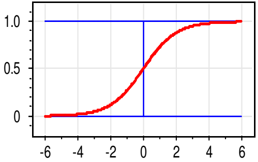
$$a(x, w) = \operatorname{sign} \sum_{j,k} w_{jk} b_{jk}(x).$$

Вес признака w_{jk} равен его вкладу в общую сумму баллов (score).

признак j	интервал k	w_{jk}
Возраст	до 25	5
	25 - 40	10
	40 - 50	15
	50 и больше	10
Собственность	владелец	20
	совладелец	15
	съемщик	10
	другое	5
Работа	руководитель	15
	менеджер среднего звена	10
	служащий	5
	другое	0
Стаж	1/безработный	0
	1..3	5
	3..10	10
	10 и больше	15
Работа_мужа /жены	нет/домохозяйка	0
	руководитель	10
	менеджер среднего звена	5
	служащий	1

Оценивание рисков в скоринге

Логистическая регрессия не только определяет веса w, но и оценивает *апостериорные вероятности* классов



$$\mathsf{P}(y|x,w) = \frac{1}{1+e^{-\langle w,x\rangle y}}$$

Оценка *риска* (математического ожидания) потерь объекта x:

$$R(x) = \sum_{y\in Y} D_{xy}\,\mathsf{P}(y|x,w),$$

где D_{xy} — величина потери для объекта x с исходом y.

Оценка говорит о том, сколько мы потеряем в среднем.
Но сколько мы потеряем в худшем случае?

Методика VaR (Value at Risk)

Стохастическое моделирование: $N = 10^4$ раз

- для каждого x_i разыгрывается исход $y_i \sim P(y|x_i)$;
- вычисляется сумма потерь по портфелю $V = \sum_{i=1}^{\ell} D_{x_i y_i}$;

99%-квантиль эмпирического распределения потерь определяет величину резервируемого капитала

Резюме в конце лекции

- Метод стохастического градиента (SG)
  — подходит для любых моделей и функций потерь
  — подходит для обучения по большим данным
- *Аппроксимация пороговой функции потерь* $L(M)$ позволяет использовать градиентную оптимизацию
- Функции $L(M)$, штрафующие за приближение к границе классов, увеличивают зазор между классами, благодаря чему повышается надёжность классификации
- *Регуляризация* снижает переобучение, возникающее в линейных моделях из-за мультиколлинеарности
- *Логистическая регрессия* — метод классификации, оценивающий условные вероятности классов $P(y|x)$