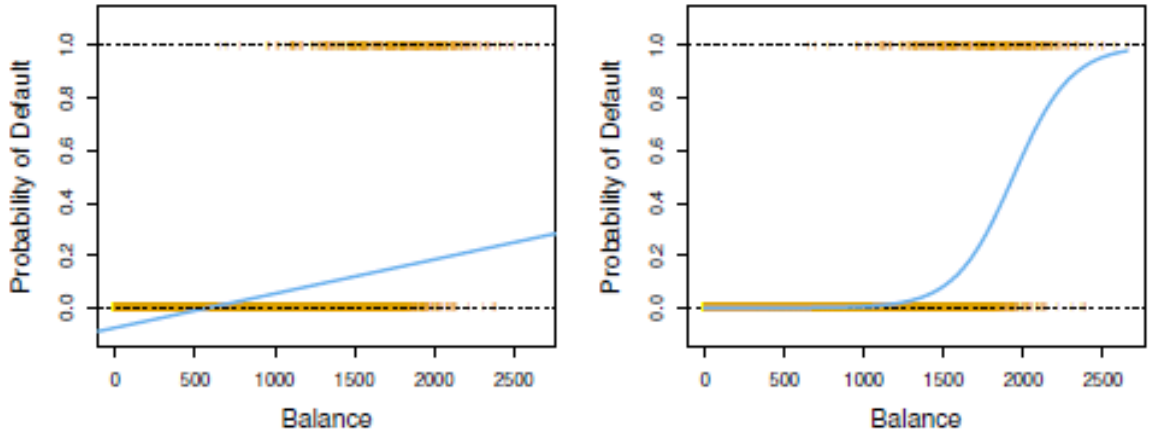
Лекция 8:
Логистическая регрессия. Задача классификации. Оценка качества моделей.

Логистическая регрессия

- Почему нельзя моделировать вероятность как непрерывный отклик с помощью линейной регрессии?



- Как представить категориальный отклик в виде числовой переменной?
- Если отклик закодирован (1=Yes, 0=No), а прогноз 1.1 или -0.4, что это означает?
- Если переменная имеет только два значения (или несколько), имеет ли смысл требовать постоянство дисперсии или нормальность ошибок?
- Вероятность ограничена, а линейная функция нет. Принимая во внимание ограниченность вероятности, можно ли предполагать линейную связь между предиктором и откликом?

Логистическая регрессия

Уравнение регрессии:

$$\text{logit}(p_i) = \mu = w_0 + w_1 x_{1i} + \cdots + w_p x_{pi}$$

Вероятность

параметр

предиктор

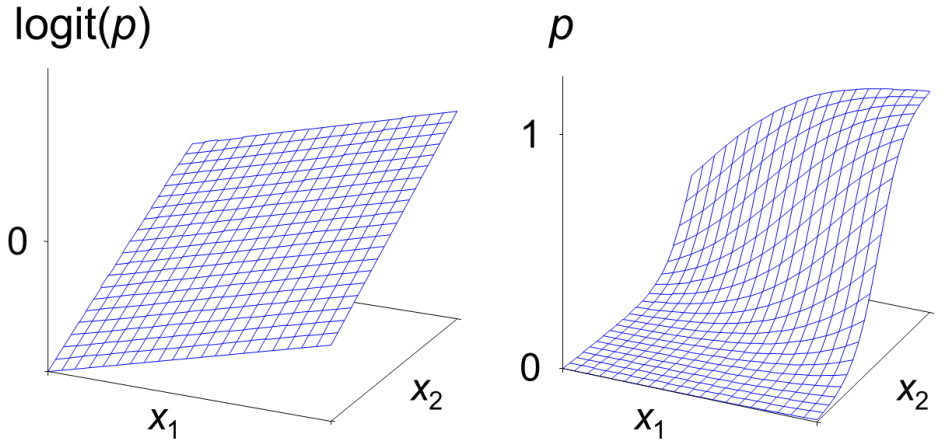
$$p_i = p(y = 1|x) = 1 - p(y = -1|x)$$

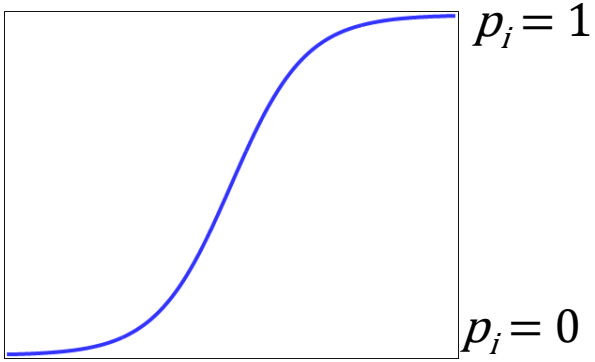
Функция связи (логит) и обратная ей (логистическая):

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right) = \mu \Rightarrow$$

$$\Rightarrow \quad p_i = \sigma(\mu) = \frac{1}{1+e^{-\mu}} = \frac{1}{1+e^{-x^T w}}$$

Основное предположение линейной логистической регрессии (линейная зависимость логита вероятности от предикторов):





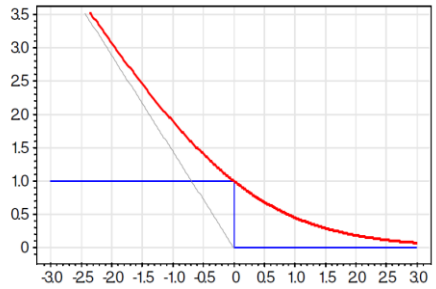
меньше $\leftarrow \mu \rightarrow$ больше

Ограничивает значение отклика

Функция потерь логистической регрессии

- **Функция потерь** (логарифмическая) является аппроксимацией негладкой функции потерь sign(.):

$$L(y, x, w) = \log\left[1 + \exp\left(-yw^T x\right)\right] \geq$$
$$\geq \text{ sign}(yw^T x)$$



- Градиент $\nabla Q(w)$ и матрица Гессе $\nabla^2 Q(w)$ для метода Ньютона-Рафсона:

$$w^{t+1} = w^t - \eta_t \left(\nabla^2 Q(w^t)\right)^{-1} \nabla Q(w^t)$$

$$\frac{\partial Q(w)}{\partial w_j} = \sum_{i=1}^{l} (1 - \sigma_i) y_i x_i, \quad \frac{\partial^2 Q(w)}{\partial w_j \partial w_k} = -\sum_{i=1}^{l} (1 - \sigma_i) \sigma_i y_i x_i x_k$$

где $\sigma_i = \sigma\left(y_i w^T x_i\right)$, $\sigma(z) = \frac{1}{1+e^{-z}}$ - сигмоидальная функция

IRLS для логистической регрессии

- На каждом шаге:
  - МНК линейной регрессии с взвешенными наблюдениями и модифицированными остатками, старающийся улучшить эмпирический риск на самых «сложных» примерах:

$$Q(w) = \sum_{i=1}^{l}(1-\sigma_i)\sigma_i\left(w^T x_i - \frac{y_i}{\sigma_i}\right)^2 \to \min_w \quad \leftrightarrow \left\|\tilde{X} - \tilde{y}w\right\|^2 \to \min_w$$

- где:
  - Взвешенная (по наблюдениям) матрица признаков $\tilde{X} = W_t X$
  - X исходная матрица данных,
  - $W_t = diag\big((1-\sigma_i)\sigma_i\big)$ – веса наблюдений на t-ой итерации,
  - поскольку $\sigma_i = P(y_i|x_i)$- вероятность правильной классификации x_i, то чем ближе x_i к границе 0.5, тем больше вес $(1-\sigma_i)\sigma_i$ и «сложнее» пример
  - $\tilde{y_i} = \frac{y_i}{\sigma_i}$ – модифицированные отклики, чем выше вероятность ошибки тем больше $\frac{1}{\sigma_i}$

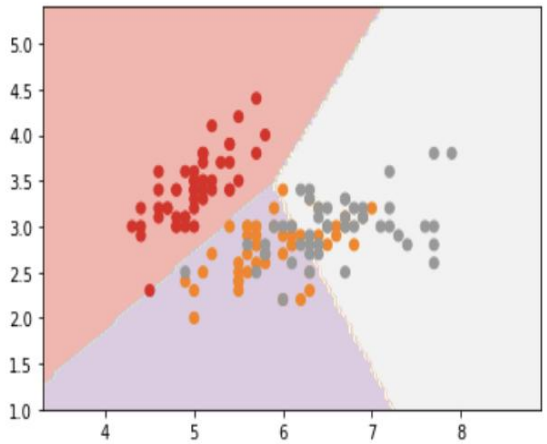
Многоклассовая логистическая регрессия и функция softmax

```
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn import datasets
from sklearn.inspection import DecisionBoundaryDisplay

iris = datasets.load_iris()
X = iris.data[:, :2]
Y = iris.target

logreg = LogisticRegression()
logreg.fit(X, Y)

DecisionBoundaryDisplay.from_estimator(
    logreg, X, cmap="Pastel1")
plt.scatter(X[:, 0], X[:, 1], c=Y, cmap="Set1")
plt.show()
```



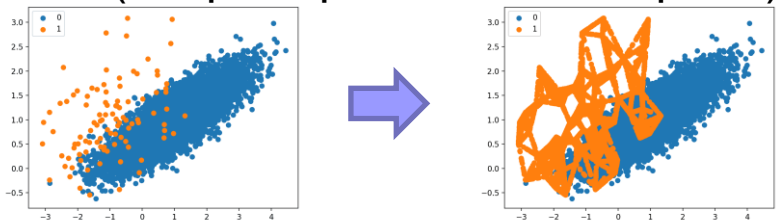
- Логистическая регрессия с двумя классами обобщается на случай K классов (многомерная логистическая функция):

$$p(y = k|x) = \frac{e^{w_k^T x}}{\sum_{j=1}^{K} e^{w_j^T x}}$$

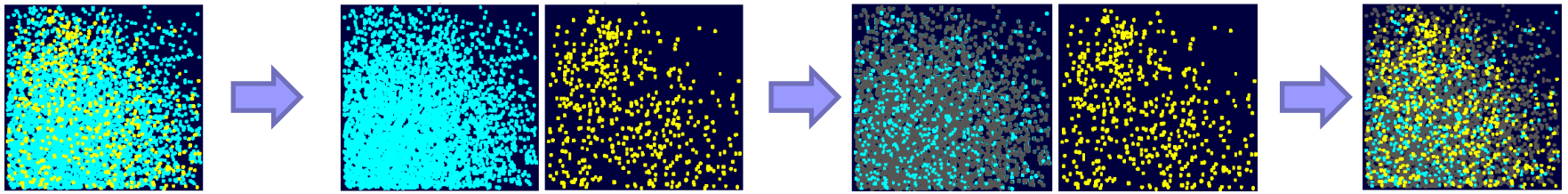
- Для *каждой* пары классов существует своя граница - линейная разделяющая функция, где вероятности классов совпадают
- Многоклассовая логистическая регрессия также называется *мультиномиальной регрессией, а* многомерная логистическая функция -softmax, *которая* «нормализует» K-мерный вектор так, чтобы сумма координат = 1

«Балансировка» выборки

- Варианты борьбы с дисбалансом:
  - Разные **веса у наблюдений** в функции потерь (обратно пропорционально общему числу наблюдений класса)
  - **Сдвиг границы** принятия решения в дискриминантной функции в сторону редкого класса пропорционально отношению размеров
  - «Балансировка» **oversampling** – с помощью некой стратегии генерируем случайные наблюдения для выборки, увеличиваем маленький класс (например, SMOTE алгоритм):



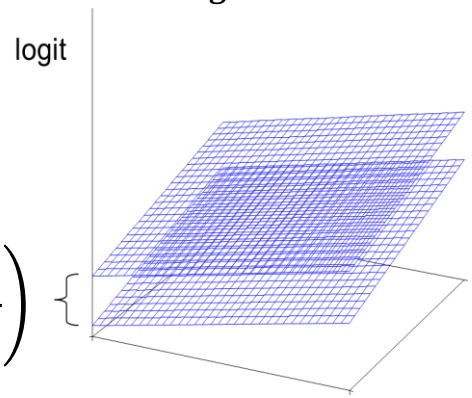
  - «Балансировка» **undersampling** – с помощью случайной выборки уменьшаем большой класс



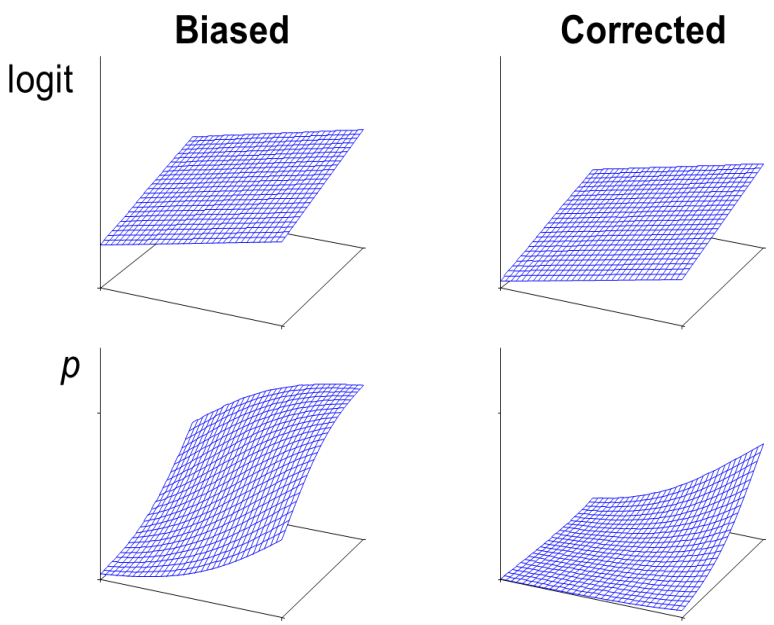
Корректировка логистической регрессии после undersamplig

- Два способа корректировки:
  - Включить параметр «сдвига» в уравнение модели

$$g(x)^{\text{adj}} = g(x)_{logit} + b$$



$$b = \log\left(\frac{\pi_0\rho_1}{\pi_1\rho_0}\right)$$



  - Скорректировать вероятности на выходе модели:

$$p_1^{adj} = \frac{p_1\pi_1\rho_0}{p_1\pi_1\rho_0 + (1-p_1)\pi_0\rho_1}$$

π_1, π_0 - до undersampling

ρ_1, ρ_0 - после undersampling

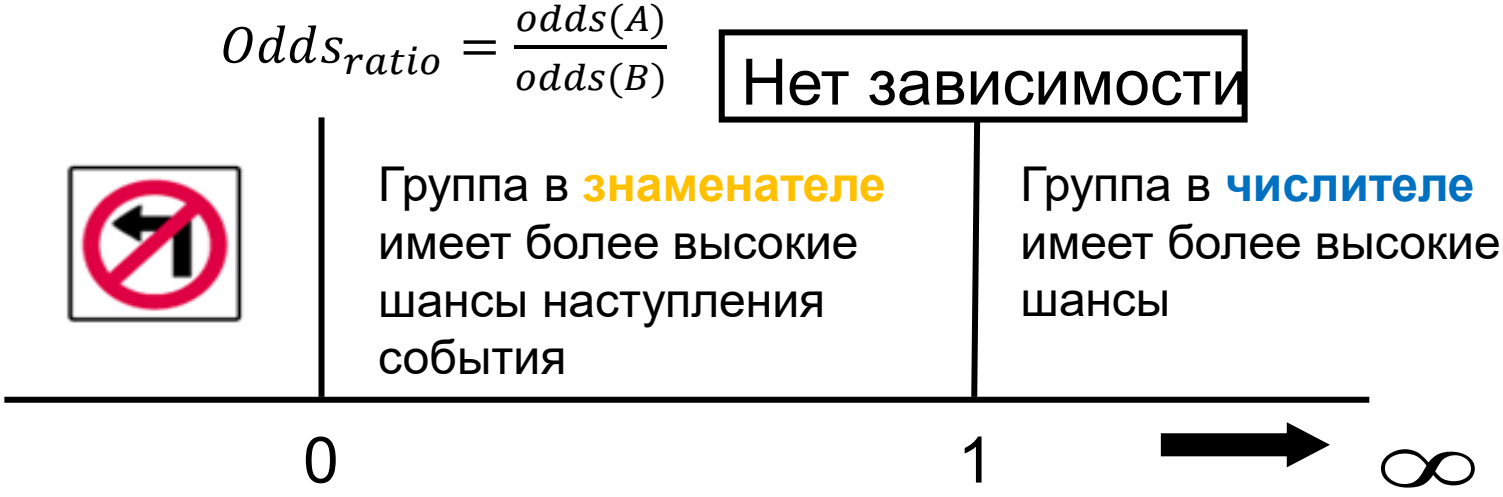
Оценка «силы» ассоциации между предиктором и бинарным откликом

- **Шанс** (это не вероятность) – отношение вероятностей события к не событию:

$$Odds = \frac{p_{event}}{p_{nonevent}}$$

- **Отношение шансов** (тоже не вероятность) показывает насколько вероятнее в терминах шансов появления события в группе A (соответствующей набору значений предикторов) по сравнению с другой группой B:

$$Odds_{ratio} = \frac{odds(A)}{odds(B)}$$



Сравнение вероятностей и шансов

	Заболел		
	Да	**Нет**	Total
Прививка	60	20	80
Без прививки	90	10	100
Total	150	30	180

Всего Заболел Без прививки ÷ Всего исходов Без прививки

Вероятность Заболел Без прививки =90÷100=0.9

Сравнение вероятностей и шансов

	Заболел		
	Да	**Нет**	Total
Прививка	60	20	80
Без прививки	90	10	100
Total	150	30	180

Вероятность Заболел Без прививки =0.90

÷

Вероятность Не заболел Без прививки =0.10

Шанс Заболеть Без прививки = 0.90÷0.10=9

Без прививки шанс заболеть в 9 раз выше чем с прививкой

Сравнение вероятностей и шансов

	Заболел		
	Да	**Нет**	Total
Прививка	60	20	80
Без прививки	90	10	100
Total	150	30	180

Шанс Заболеть с прививкой=3

÷

Шанс Заболеть Без прививки=9

Отношение шансов =3÷9=0.3333

Шансов заболеть с прививкой в 3 раза меньше чем без

Отношение шансов в логистической регрессии

- Используется для оценки влияния переменной на отклик и показывает как изменятся шансы при изменении i-ой переменной на 1 (равно exp от коэффициента):

$$\text{logit}(p) = \log(odds) = w_0 + w_i x_i + \sum_{j \neq i} w_j x_j \Rightarrow$$

$$odds = \exp(w_0 + w_i x_i + \sum_{j \neq i} w_j x_j)$$

$$\text{logit}(p`) = \log(odds`) = w_0 + w_i(x_i + 1) + \sum_{j \neq i} w_j x_j \Rightarrow$$

$$odds` = \exp(w_0 + w_i(x_i + 1) + \sum_{j \neq i} w_j x_j)$$

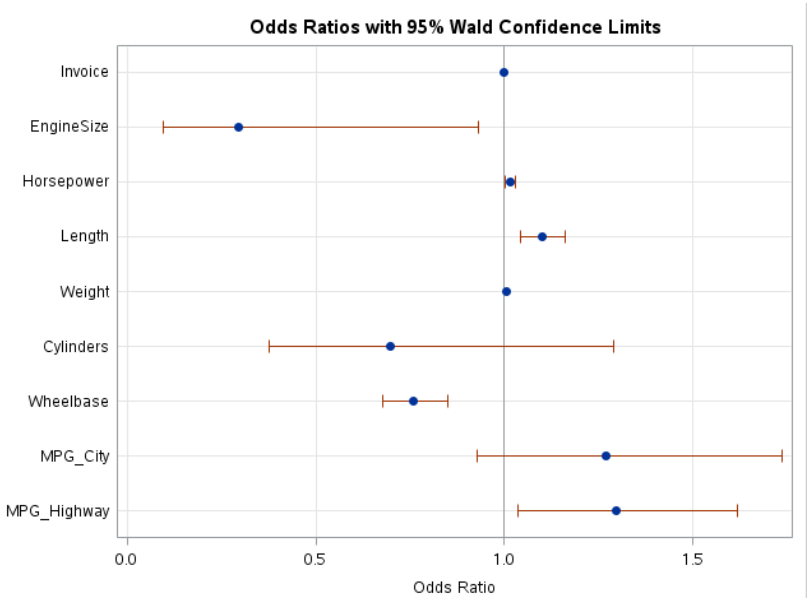
$$odds_{ratio} = \frac{odds'}{odds} = \exp(w_i)$$

- Если больше 1 – шансы увеличиваются, если меньше, то уменьшаются, интерпретация как в пуассоновской регрессии

Отношение шансов и важность переменных

exp(.)

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Invoice	1.000	1.000	1.000
EngineSize	0.295	0.094	0.931
Horsepower	1.016	1.003	1.029
Length	1.100	1.044	1.160
Weight	1.005	1.004	1.007
Cylinders	0.696	0.376	1.289
Wheelbase	0.757	0.676	0.849
MPG_City	1.270	0.929	1.736
MPG_Highway	1.295	1.036	1.618



Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-10.7686	4.6784	5.2983	0.0213
Invoice	1	-0.00013	0.000028	21.9445	<.0001
EngineSize	1	-1.2200	0.5858	4.3365	0.0373
Horsepower	1	0.0156	0.00666	5.4867	0.0192
Length	1	0.0957	0.0270	12.6146	0.0004
Weight	1	0.00529	0.000908	33.9767	<.0001
Cylinders	1	-0.3625	0.3146	1.3275	0.2493
Wheelbase	1	-0.2778	0.0580	22.9685	<.0001
MPG_City	1	0.2389	0.1595	2.2421	0.1343
MPG_Highway	1	0.2584	0.1136	5.1710	0.0230

- Можно найти не только точечную оценку ОШ (OR), но и доверительный интервал
- Если он содержит 1, то доверительный интервал коэффициента содержит 0, т.е. предиктор не значимый
- Не учитывается разброс переменной

Категориальные предикторы

- Схемы кодировки:
  - Effect coding (относительно «среднего»)

Переменная	Значение	Обозначение	1	2
IncLevel	1	Low Income	1	0
	2	Medium Income	0	1
	3	High Income	-1	-1

  - Reference coding (относительно «базового»)

Переменная	Значение	Обозначение	1	2
IncLevel	1	Low Income	1	0
	2	Medium Income	0	1
	3	High Income	0	0

Effect coding: Пример

$$\text{logit}(p) = w_0 + w_1 * D_{\text{Low income}} + w_2 * D_{\text{Medium income}}$$

w_0= Общий логарифм от шанса по всем категориям

w_1= разница между логарифмом шанса Low income и w_0

w_2 = разница между логарифмом шанса Medium income и общим

Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-0.5363	0.1015	27.9143	<.0001
IncLevel	**1**	1	-0.2259	0.1481	2.3247	0.1273
IncLevel	**2**	1	-0.2200	0.1447	2.3111	0.1285

Reference coding: Пример

$$\text{logit}(p) = w_0 + w_1 * D_{\text{Low income}} + w_2 * D_{\text{Medium income}}$$

w_0 = Логарифм шанса для High

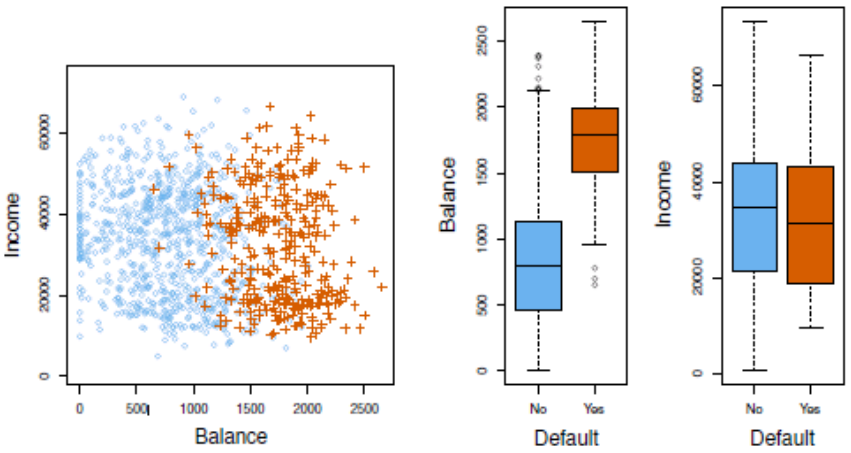
w_1 = Разница между логарифмами шанса Low и High

w_2 = Разница между логарифмами шанса между Medium и High

Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-0.0904	0.1608	0.3159	0.5741
IncLevel	**1**	1	-0.6717	0.2465	7.4242	0.0064
IncLevel	**2**	1	-0.6659	0.2404	7.6722	0.0056

Задача классификации

- Переменная отклика является *категориальной,* например:
  - *Бинарная классификация:* почтовое сообщение может принадлежать одному из следующих классов Y = (spam; ham)
  - *Многоклассовая классификация:* изображение цифры принадлежит одному из классов $\mathrm{Y} = \{0, \ldots, 9\}$
- Цели:
  - Построить классификатор $a: X \to Y$, который связывает метку класса с неразмеченным x
  - Понимание роли различных предикторов $x = (x_1, \ldots, x_p)$
  - Оценка достижимого **качества** классификации



Задача классификации

- Дано: множество «размеченных» примеров :
  - обучающая выборка или тренировочный набор:
    $$Z = \{(x_i, y_i)\}_{i=1}^{l}$$
  - $y_i \in Y = \{C_1, \dots, C_k\}$: известный **категориальный** отклик и его множество значений мощности *K*
- Основные определения:
  - Постановка задачи: найти алгоритм (или гипотезу, или модель)
    $$a_Z : X \to Y$$
  - Естественная функция потерь – несовпадение прогноза (неудобно - не дифференцируема):
    $$L(x, y) = [y \neq a(x)]$$

Дискриминантные функции

- Во многих случаях удобно пользоваться дискриминантными функциями для каждого класса c: $g_c: X \to G \subset \mathbb{R}$, такими что:

$$a^*(x) = \operatorname*{argmax}_c g_c(x)$$

- Частный (и частый) случай *байесовский классификатор*:

$$g_c(x) = P(y = c|x),\ G = [0,1]$$

- Граница между классами i и j:

$$\{x \in X | g_i(x) = g_j(x)\}$$

- Отступ оценивает «гладкое» качество классификации:

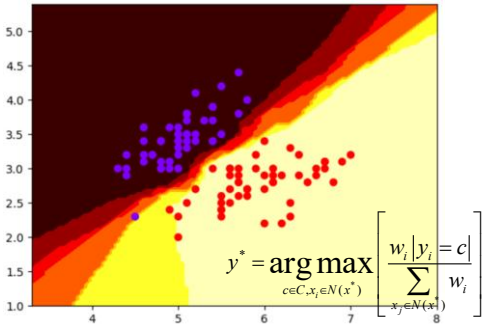
$$M(x, y) = g_y(x) - \max_{c \neq y} g_c(x)$$

- Линейный классификатор:

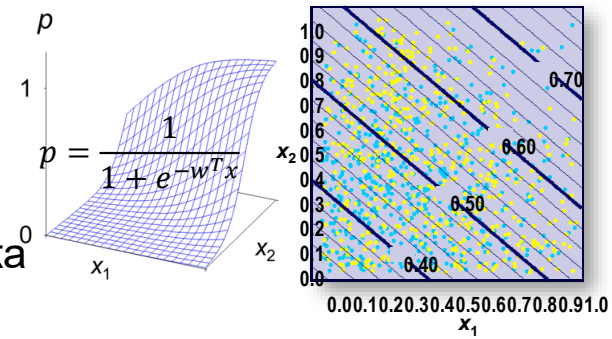
$$g_c(x) = \langle w_c, x \rangle, \text{ где вектор } x = [x_1, \dots, x_p, 1]$$

Примеры методов классификации

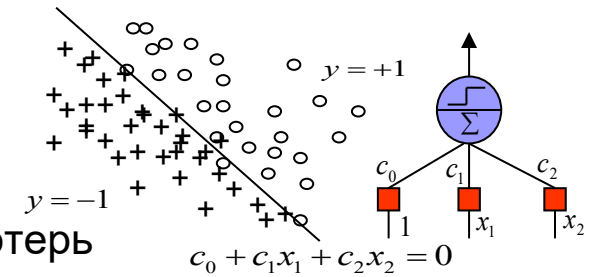
- KNN (не линейный):
  - Взвешенные, kernel и др.
  - Прогноз – голосование соседей по окрестности
  - Дискр. ф-ция – усреднение голосов
  - Обучения нет, вместо него поиск



- Логистическая регрессия (линейная):
  - GLM с распределением Бернулли
  - Прогноз – вероятность целевого отклика
  - Дискр. ф-ция – линейная $w^T x$
  - Обучение – методы оптимизации 1 или 2 порядка



- Линейные гиперплоскости (персептроны):
  - Разделяющие гиперплоскости
  - Прогноз – знак в уравнении гиперплоскости
  - Дискр. ф-ция – расстояние до гиперплоскости
  - Обучение – например, SGD, разные функции потерь



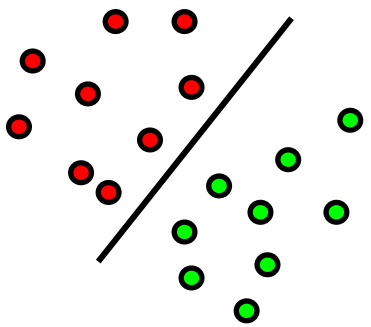
Бинарный классификатор

- Два класса $|Y| = 2$
  - например $Y = \{0,1\}$, $Y = \{-, +\}$, $Y = \{no, yes\}$
  - один класс – целевой, или класс «событие», обычно 1, +, yes
  - другой класс - не целевой или «не событие», обычно 0, -, no
- Дискриминантная функция, отступ и модель, примеры:
  - Разделяющая гиперплоскость:

    $Y = \{-1, +1\}$, $g(x) = g_+(x) - g_-(x)$,

    $M(x, y) = yg(x)$,

    $a(x) = sign(g(x))$



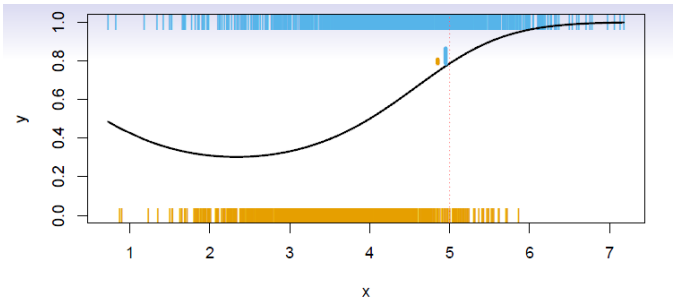
  - Вероятностная модель:

    $Y = \{0,1\}$, $g(x) = p(y = 1|x)$,

    $M(x, y) = p(y = 1|x) - p(y = 0|x) =$

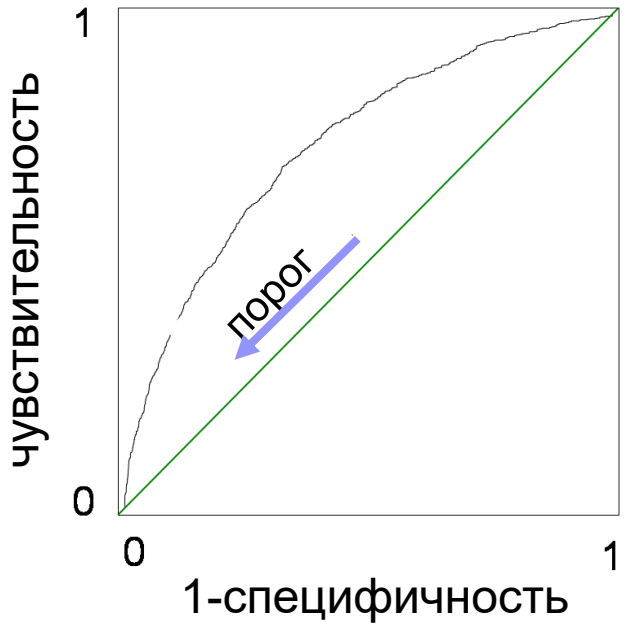
    $= 2p(y = 1|x) - 1$,

    $a(x) = [M(x)]_+$



Оценка качества бинарного классификатора

Реальность \ Прогноз	0	1	
0	True Negative	False Positive	Всего 0 (не событий)
1	False Negative	True Positive	Всего 1 (событий)
	Всего прогноз 0 (не событий)	Всего прогноз 1 (событий)	



ЧУВСВИТЕЛЬНОСТЬ (true positive rate (TPR), hit rate, recall)
TPR = R = SE = TP / (TP+FN)

СПЕЦИФИЧНОСТЬ (true negative rate (TNR))
SPC = TN / (FP + TN)

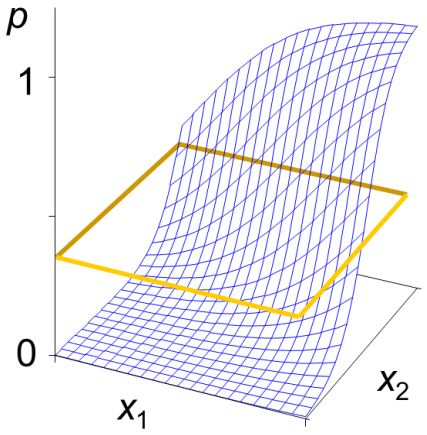
«АККУРАТНОСТЬ»
ACC= (TN+TP) / (FP + FN+TP+TN)

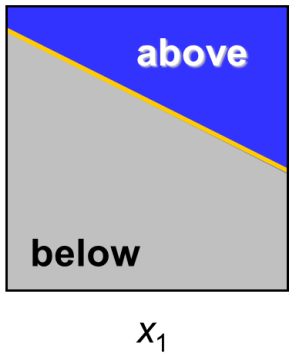
F-МЕРА(гарм. среднее)
$F_b = (1-b^2)*P*R/(b^2*(P+R))$

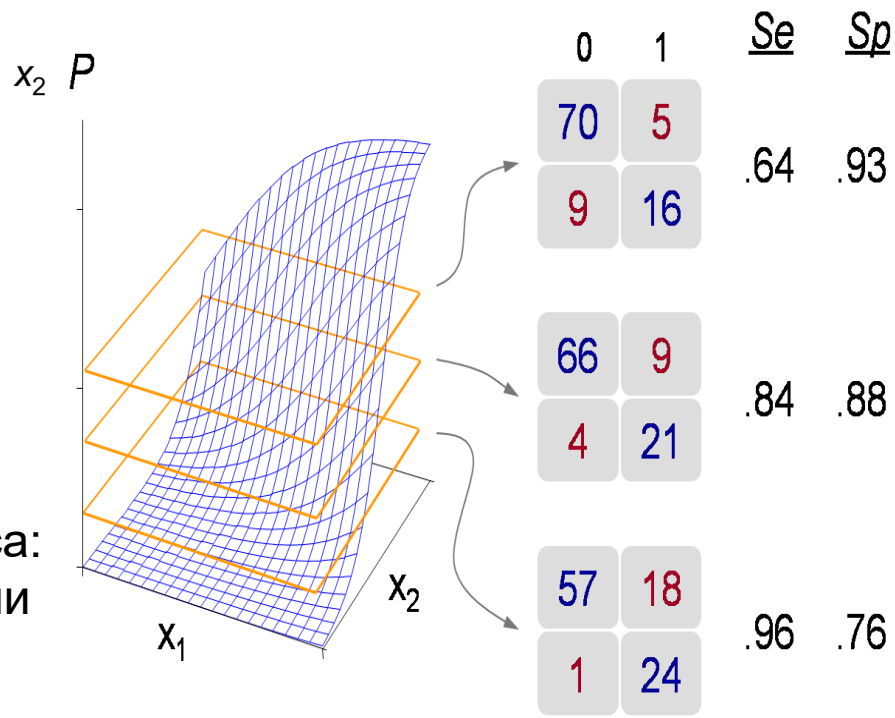
УРОВЕНЬ ОШИБОК
ERR = 1-ACC

ТОЧНОСТЬ (Precision)
P= TP / (TP+FP)

Оценка бинарного классификатора







Матрица выигрыша-проигрыша:

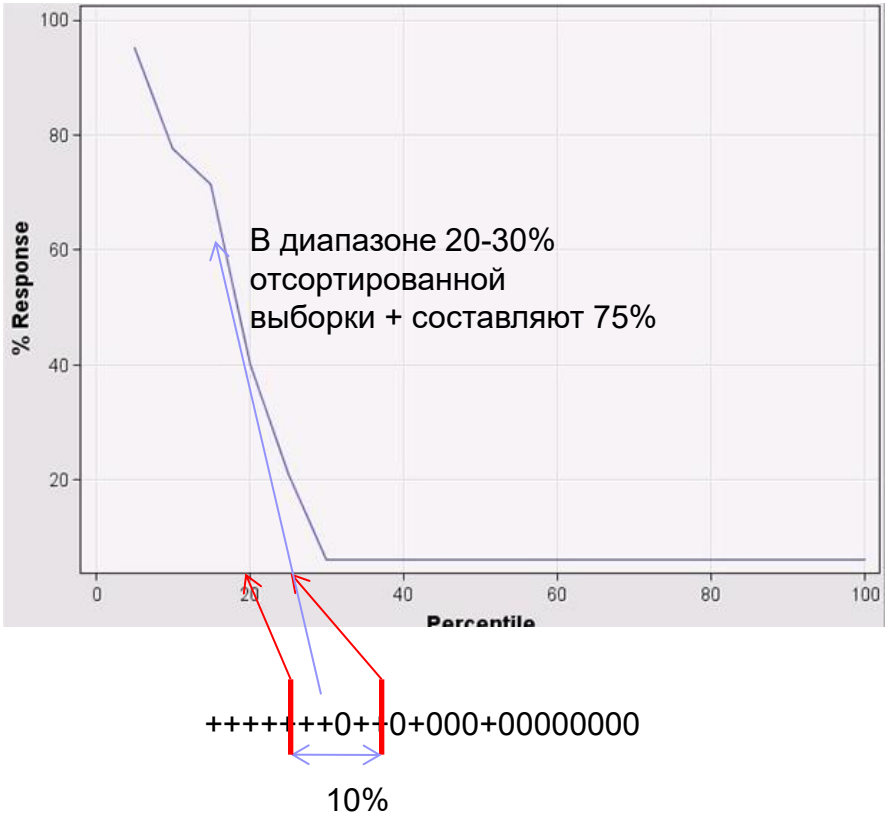
Реальность \ решение	0	1
0	δ_{TN}	δ_{FP}
1	δ_{FN}	δ_{TP}

Правило Байеса:
Решение 1 если

$$P > \frac{1}{1+\left(\frac{\delta_{TP}-\delta_{FN}}{\delta_{TN}-\delta_{FP}}\right)}$$

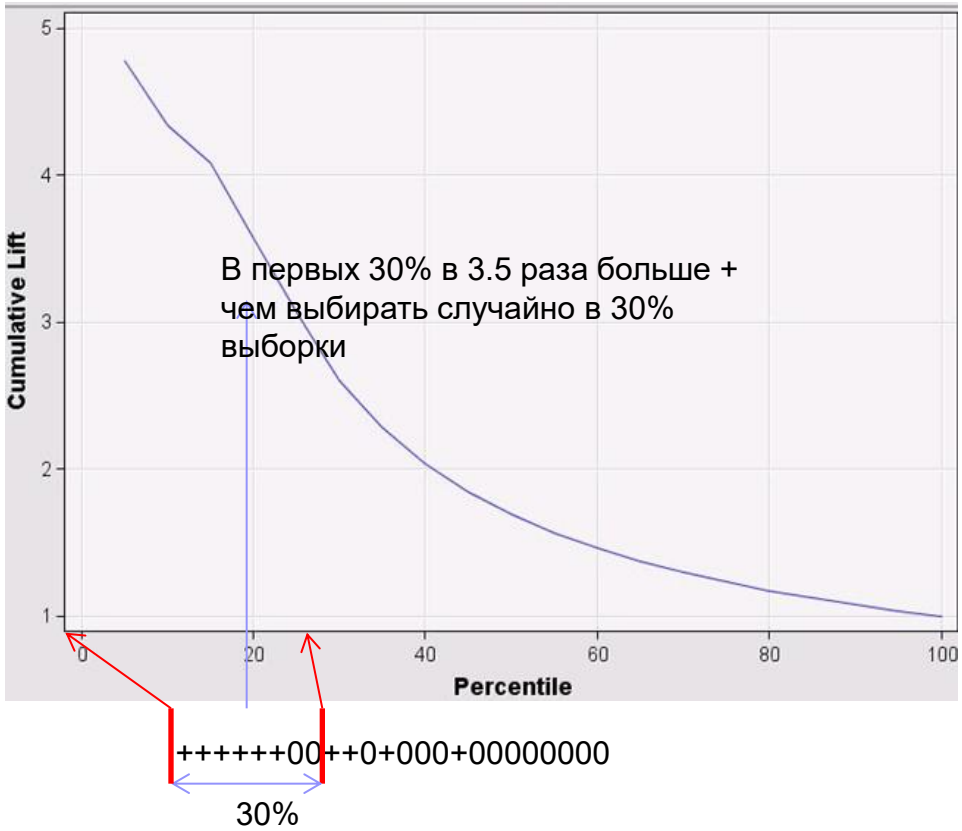
Графические средства сравнения моделей: Response (отклик)

- Процедура построения:
  - Сортируем (например, слева направо) набор по убыванию дискриминантной функции
  - Идем порогом отсечения по отсортированному набору (слева направо) с некоторым диапазоном (как правило кратно 5%)
  - Для каждого положения диапазона считаем отношение числа положительных примеров к числу всех примеров внутри диапазона
  - Ставим точку на графике
- Агрегированный отклик (cumulative response):
  - Диапазон всегда с 0



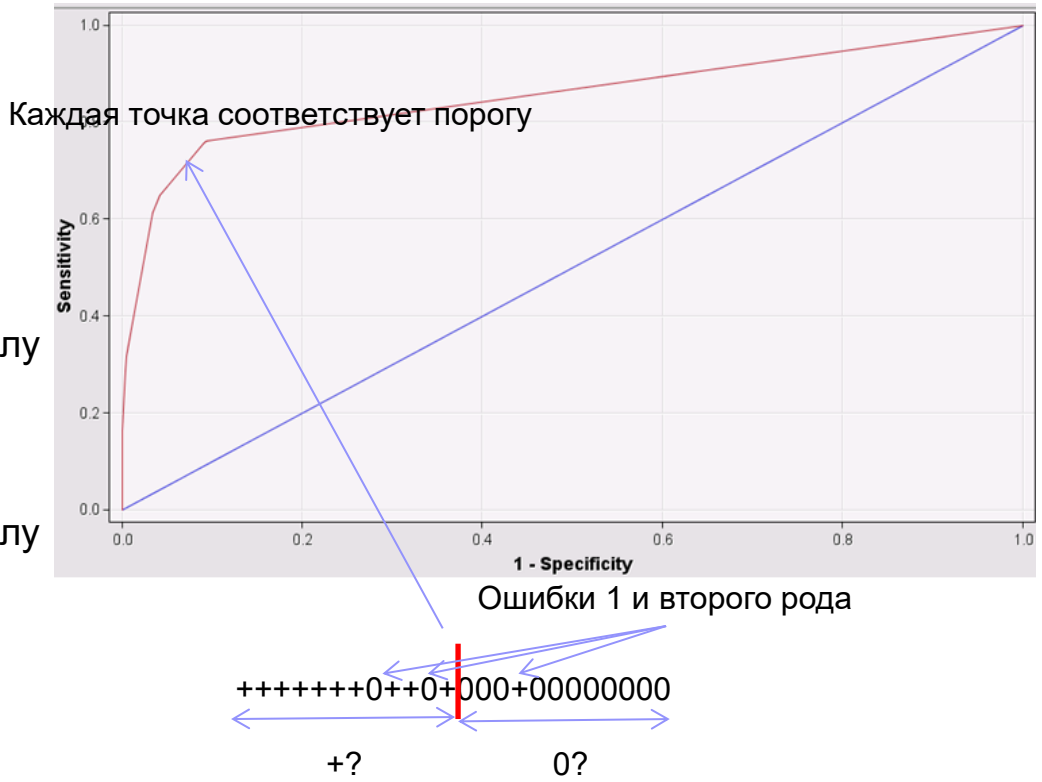
Графические средства сравнения моделей: Lift (подъем)

- Процедура построения:
  - Сортируем (например, слева направо) набор по убыванию дискриминантной функции
  - Идем порогом отсечения по отсортированному набору (слева направо) с некоторым диапазоном (как правило кратно 5%)
  - Для каждого положения диапазона считаем отношение числа положительных примеров к числу положительных примеров, которые могли бы быть выбраны «случайно» - без модели
  - Ставим точку на графике
- Агрегированный подъем(cumulative lift):
  - Диапазон всегда с 0



ROC кривая

- Процедура построения:
  - Сортируем набор по убыванию дискриминантной функции
  - Идем порогом отсечения по отсортированному набору
  - Для каждого положения порога считаем:
    1. отношение числа положительных примеров «слева» от порога к числу всех положительных примеров – true positive rate
    2. отношение числа отрицательных примеров «слева» от порога к числу всех отрицательных примеров – false positive rate
  - Ставим точку на графике



Оценка на основе согласованности всевозможных пар наблюдений (правильной упорядоченности наблюдений в паре), принадлежащих разным классам.

AUC – мера согласованности

- Дано:
  - бинарная задача классификации в постановке $Y = \{-1, +1\}$
  - $x_{(1)}, \dots, x_{(l)}$- упорядочены по $g(x_{(i)}) \leq g(x_{(i+1)})$
- Поскольку:

$$TPR_k = \frac{1}{l_+}\sum_{i=k}^{l}[y_{(i)} = +1],\ FPR_k = \frac{1}{l_-}\sum_{i=k}^{l}[y_{(i)} = -1]$$

- Получаем AUC:
  - вероятность правильно упорядочить пары наблюдений из разных классов (формула трапеции вычисления интеграла):

$$AUC = \sum_{k=1}^{l-1}\frac{TPR_{k+1} + TPR_k}{2}(FPR_k - FPR_{k+1}) =$$

$$= \frac{1}{l_- l_+}\sum_{k=1}^{l-1}\sum_{i=k+1}^{l}[y_{(i)} = +1][y_{(k)} = -1] = \frac{1}{l_- l_+}\sum_{k<i}[y_{(i)} > y_{(k)}]$$

  - не все однозначно с «ничьими», обычно их берут с весом 0.5, но есть разные подходы (в том числе 0 – считать ничьи несогласованными, 1 – считать ничьи согласованными)

Максимизация AUC напрямую с помощью линейной модели

- Дано:
  - бинарная задача классификации в постановке $Y = \{-1, +1\}$
  - Модель классификации в классе $a(x, w) = sign(g(x, w))$
- Поскольку AUC – доля правильно упорядоченных пар из разных классов, то:

$$AUC(w) = \frac{1}{l_- l_+} \sum_{i,j} [y_i < y_j](g(x_i, w) - g(x_j, w)) \to \max_w$$

- Эмпирический риск (функция потерь ранжирования):

$$1 - AUC(w) \le Q(w) = \frac{1}{l_- l_+} \sum_{i,j: y_i < y_j} \underbrace{L(g(x_i, w) - g(x_j, w))}_{\text{Отступ для пары } x_i, x_j:\ M(x_i, x_j, w)} \to \min_w$$

- Алгоритм – например, SGD

Логистическая функция потерь для оценки качества вероятностных моделей

- AUC инвариантна к монотонному преобразованию отклика – не лучший вариант для вероятностных классификаторов
- Поэтому используют логарифмическое правдоподобие для распределения Бернулли
- Если $Y = \{0,1\}, Z = \{(x_i, y_i)\}_{i=1}^{l}$, $p(x) = a(x) = \underset{y \in Y}{\operatorname{argmax}} P(y|x)$:

$$logloss(Z, p(x)) = -\frac{1}{l}\sum_{i=1}^{l}\left[y_i \log\left(p(x_i)\right) + (1 - y_i)\log(1 - p(x_i))\right]$$

- Ключевой недостаток:
  - чувствительность к «грубым» ошибкам
  - неограниченный рост потерь
  - делают «пороги толерантности»
- Многоклассовое (C классов) обобщение:

$$logloss(Z, p(x)) = -\frac{1}{lC}\sum_{i=1}^{l}\sum_{j=1}^{C} y_{ij} \log\left(p_j(x_i)\right)$$

y_{ij} - бинарный OneHotEncoding метки класса