Обучение с учителем. Классификация. Дискриминантный анализ. Логистическая регрессия. Метод опорных векторов. Выбор модели с помощью кросс-валидации. Метод стохастического градиента

Белкова Анна, Редкокош Кирилл, Лобанова Полина

гр. 21.М03-мм
Санкт-Петербургский государственный университет
Кафедра статистического моделирования

11 декабря 2022 г.

Метрики качества классификации. Матрица ошибок

Обсудим распространённые подходы к измерению качества моделей.
Допустим, что у нас есть два класса и алгоритм, предсказывающий принадлежность каждого объекта одному из классов, тогда матрица ошибок классификации будет выглядеть следующим образом:

	y=1	y=0
$\hat{y} = 1$	True Positive (TP)	False Positive (FP)
$\hat{y} = 0$	False Negative (FN)	True Negative (TN)

Где $\hat{y}$ — это ответ алгоритма на объекте, а y — истинная метка класса на этом объекте. Таким образом, ошибки классификации бывают двух видов: False Negative (FN) и False Positive (FP).

Accuracy

Интуитивно понятной, очевидной и почти неиспользуемой метрикой является accuracy — доля правильных ответов алгоритма:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Эта метрика бесполезна в задачах с неравными классами, и это легко показать на примере.

Precision, recall

Для оценки качества работы алгоритма на каждом из классов по отдельности введем метрики precision (точность) и recall (полнота).

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Precision можно интерпретировать как долю объектов, названных классификатором положительными и при этом действительно являющимися положительными, а recall показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел алгоритм

Precision, recall

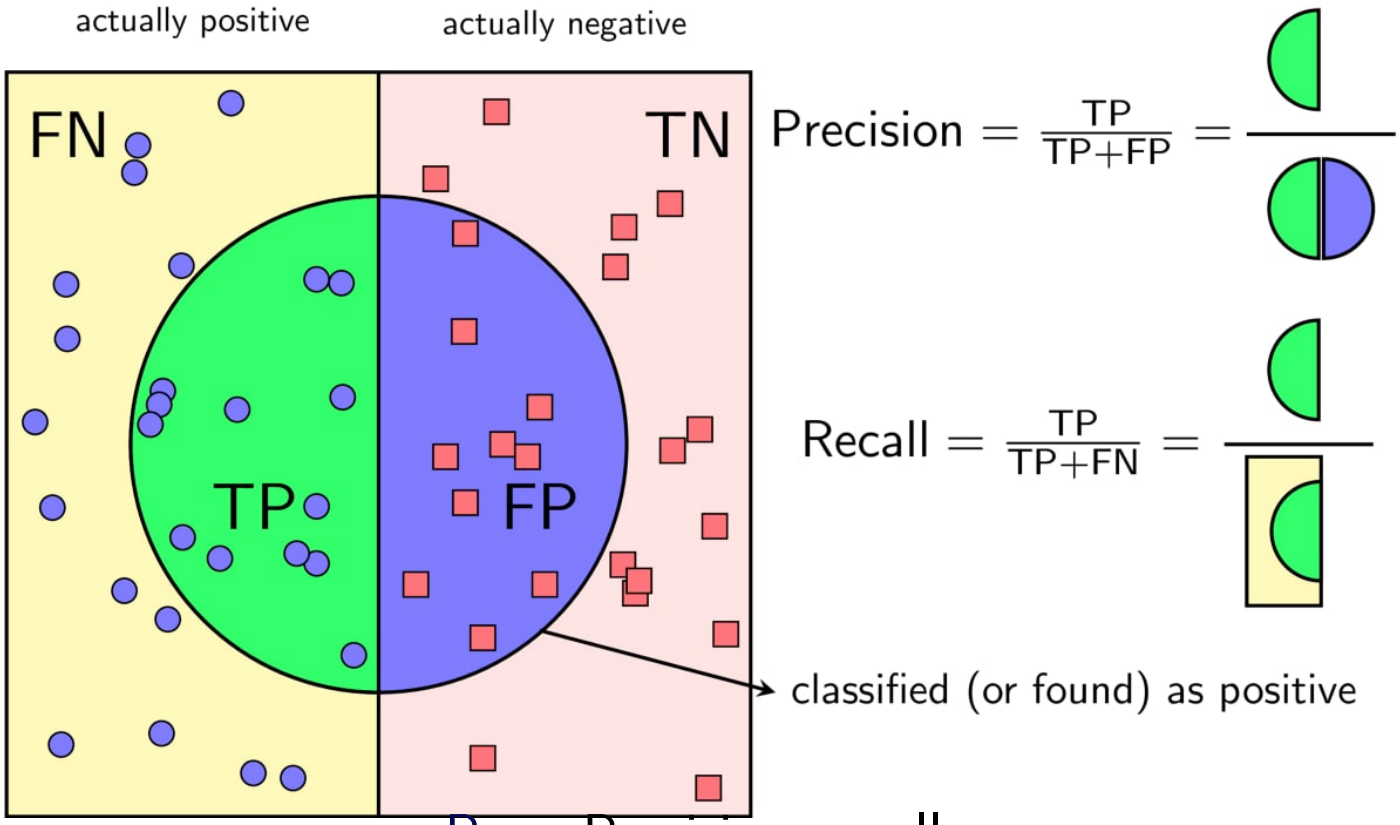


Рис.: Precision, recall

F-мера

Существует несколько различных способов объединить precision и recall в агрегированный критерий качества. F-мера (в общем случае F_β) — среднее гармоническое precision и recall :

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

β в данном случае определяет вес точности в метрике, и при $\beta = 1$ это среднее гармоническое (с множителем 2, чтобы в случае precision = 1 и recall = 1 иметь $F_1 = 1$) F-мера достигает максимума при полноте и точности, равными единице, и близка к нулю, если один из аргументов близок к нулю.

AUC-ROC

Одним из способов оценить модель, является AUC-ROC (или ROC AUC) — площадь (Area Under Curve) под кривой ошибок (Receiver Operating Characteristic curve). Данная кривая представляет из себя линию от (0,0) до (1,1) в координатах True Positive Rate (TPR) и False Positive Rate (FPR):

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

AUC-ROC

TPR– это полнота, а FPR показывает, какую долю из объектов negative класса алгоритм предсказал неверно. В идеальном случае, когда классификатор не делает ошибок (FPR = 0, TPR = 1) мы получим площадь под кривой, равную единице; в противном случае, когда классификатор случайно выдает вероятности классов, AUC-ROC будет стремиться к 0.5, так как классификатор будет выдавать одинаковое количество TP и FP.

AUC-ROC

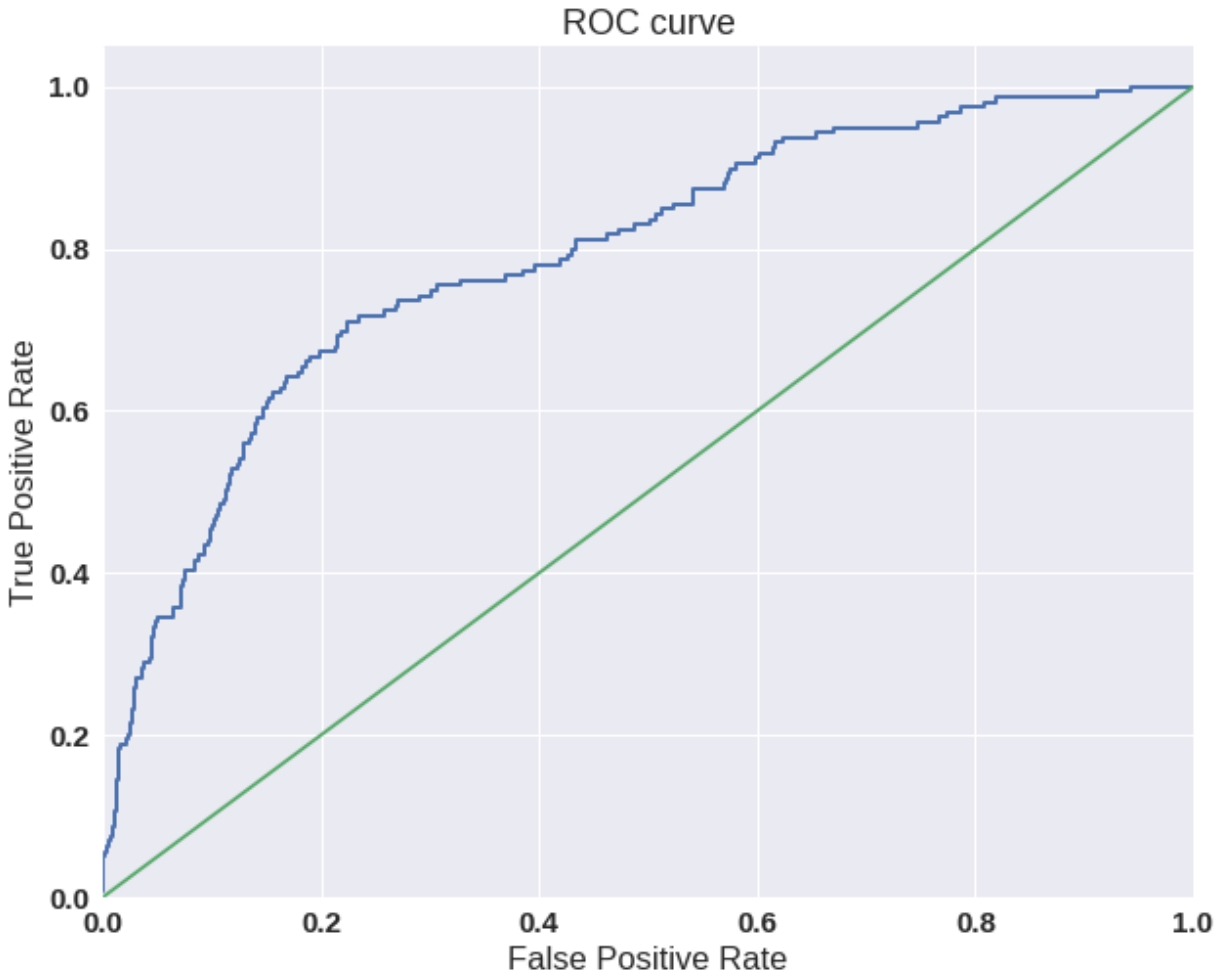


Рис.: ROC-кривая

AUC-PR

Precision и recall также используют для построения кривой и, аналогично AUC-ROC, находят площадь под ней:

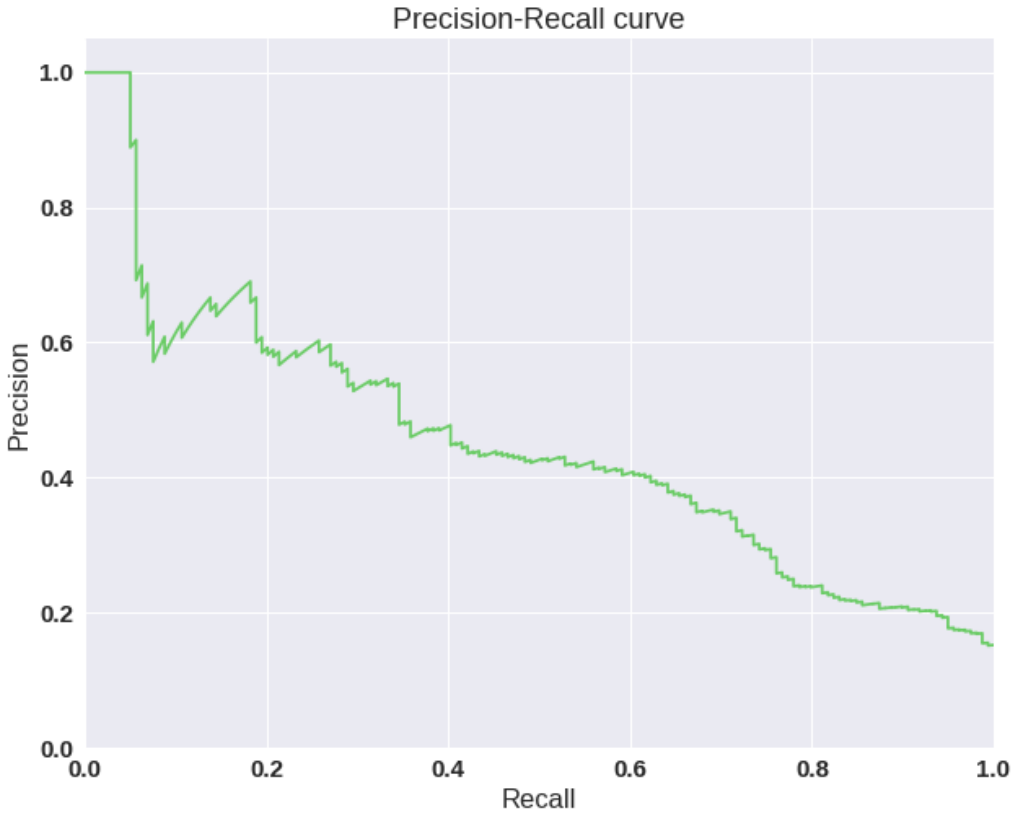
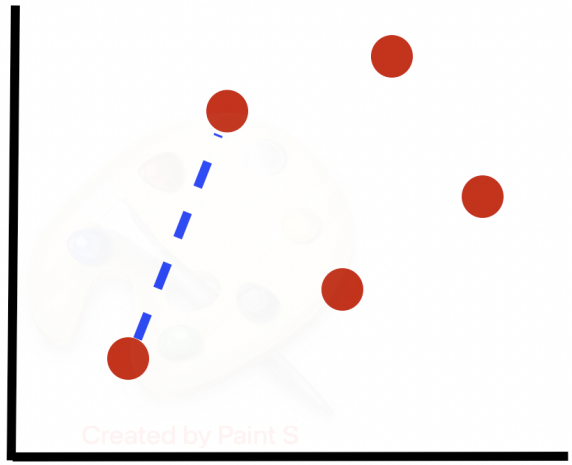


Рис.: PR-кривая

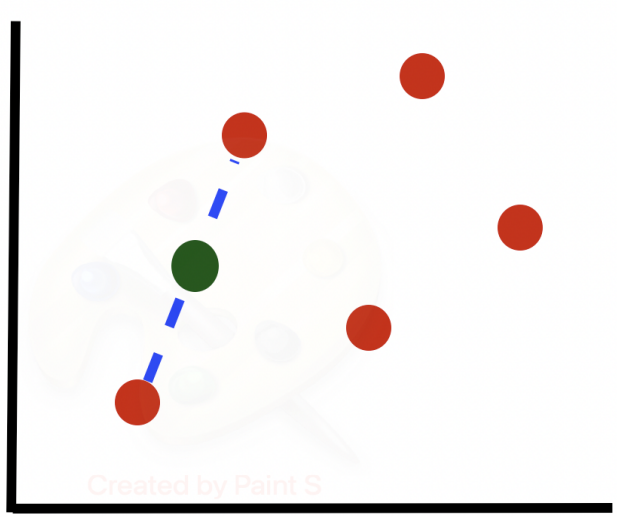
Модификации датасета для выравнивания соотношения классов

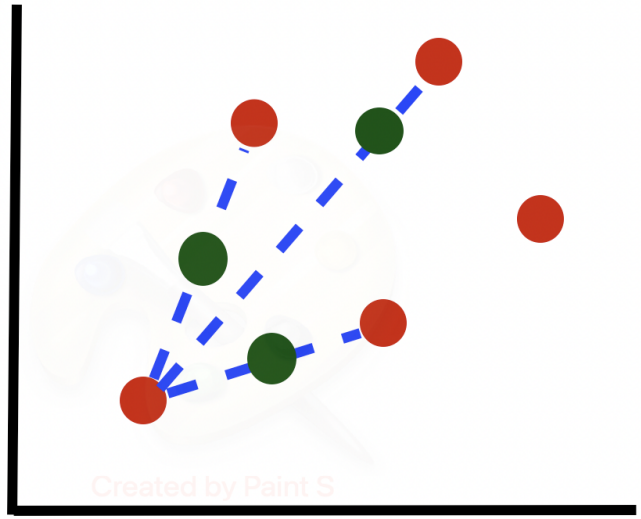
Одним из распространенных способов решения проблемы несбалансированных данных является избыточная выборка. Чрезмерная выборка относится к различным методам, которые направлены на увеличение количества экземпляров из недопредставленного класса в наборе данных.
Самый простой способ сделать это - случайным образом выбрать наблюдения из класса меньшинства и добавить их в набор данных, пока мы не достигнем баланса между большинством и классом меньшинства.

(Synthetic Minority Over-sampling Technique, SMOTE)



Step 1

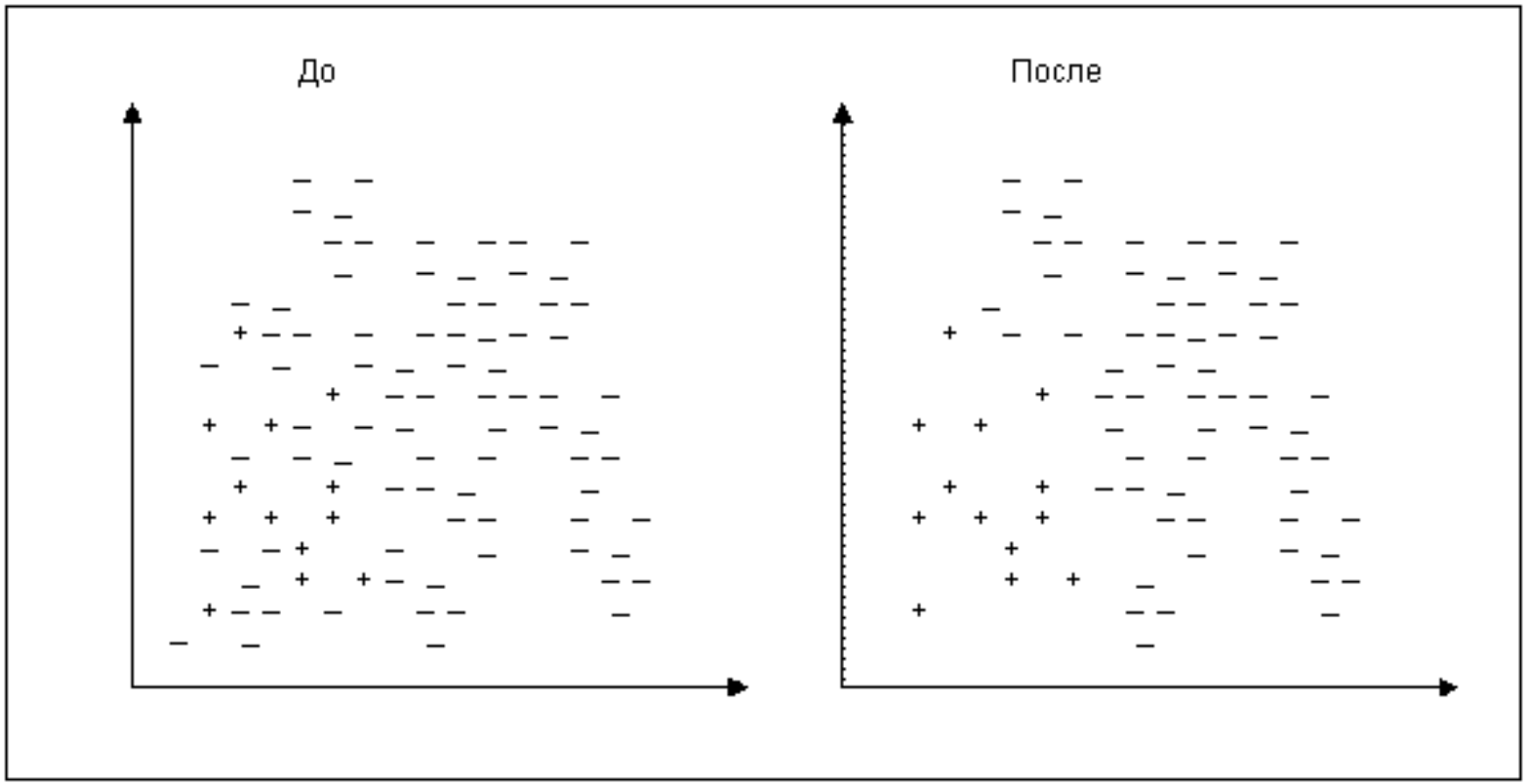




SMOTE Расширения

- BorderlineSMOTE: Вместо избыточной выборки между всеми наблюдениями меньшинств, BorderlineSMOTE стремится увеличить количество наблюдений меньшинств, которые граничат с наблюдениями большинства. Цель здесь - дать классификатору возможность более четко различать эти пограничные наблюдения.
- SVMSMOTE: SVMSMOTE, как следует из его названия, использует алгоритм машины опорных векторов для генерации новых наблюдений меньшинства вблизи границы между классами большинства и меньшинства.

Tomek Links



Дискриминантный анализ

Суть дискриминантного анализа заключается в том, чтобы смоделировать распределение X в каждом из классов отдельно, а затем использовать теорему Байеса, чтобы получить $P(Y = k \mid X = x)$.

Дискриминантный анализ

Суть дискриминантного анализа заключается в том, чтобы смоделировать распределение X в каждом из классов отдельно, а затем использовать теорему Байеса, чтобы получить $P(Y = k \mid X = x)$.
Теорема Байеса:

$$P(Y = k \mid X = x) = \frac{P(X = x \mid Y = k) \cdot P(Y = k)}{P(X = x)}.$$

Байесовский классификатор

Для построения байесовского классификатора, нам необходимо знать апостериорные вероятности $P(Y \mid \xi = x)$.
Обозначим $p_i(x) = P(\xi = x \mid \eta = Y_i)$ условные плотности классов, $\pi_i = P(\eta = Y_i)$ – априорные вероятности, $\sum_{i=1}^{K} \pi_i = 1$.

Байесовский классификатор

Для построения байесовского классификатора, нам необходимо знать апостериорные вероятности $P(Y \mid \xi = x)$.
Обозначим $p_i(x) = P(\xi = x \mid \eta = Y_i)$ условные плотности классов, $\pi_i = P(\eta = Y_i)$ – априорные вероятности, $\sum_{i=1}^{K} \pi_i = 1$.

По теореме Байеса получим:

$$P(Y = i \mid X = x) = \frac{p_i(x)\pi_i}{\sum_{i=1}^{K} p_i(x)\pi_i}.$$

Поэтому в качестве классифицирующих функций берут

$$f_i(x) = \mathrm{P}(x|C_i)\,\pi_i = p_i(x)\pi_i.$$

LDA

Предполагаем, что классы имеют нормальное распределение с одинаковой ковариационной матрицей.
Тогда плотность в точке x:

$$p_i(x) = p\left(x|\xi = Y_i\right) = \frac{1}{(2\pi)^{p/2}\left|\mathbf{\Sigma}\right|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T\mathbf{\Sigma}^{-1}(x-\mu_i)\right)$$

LDA

Предполагаем, что классы имеют нормальное распределение с одинаковой ковариационной матрицей.
Тогда плотность в точке x:

$$p_i(x) = p\left(x|\xi = Y_i\right) = \frac{1}{(2\pi)^{p/2}\left|\mathbf{\Sigma}\right|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T \mathbf{\Sigma}^{-1}(x-\mu_i)\right)$$

И классифицирующая функция $f_i(x) = \pi_i p\left(x|\xi = Y_i\right)$, где π_i — априорная вероятность наблюдения попасть в i-ю группу.

LDA

Для упрощения вычислений можно переписать классифицирующую функцию через возрастающее монотонное преобразование как:

$$g_i(x) = \log f_i(x) = \log \pi_i - \frac{1}{2}\log|\mathbf{\Sigma}| - \frac{1}{2}(x-\mu_i)^T \mathbf{\Sigma}^{-1}(x-\mu_i).$$

LDA

Для упрощения вычислений можно переписать классифицирующую функцию через возрастающее монотонное преобразование как:

$$g_i(x) = \log f_i(x) = \log \pi_i - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i).$$

Сократив часть, не зависящую от номера класса, получаем линейные классифицирующие функции:

$$h_i(x) = -\frac{1}{2}\boldsymbol{\mu}_i{}^T\Sigma^{-1}\boldsymbol{\mu}_i + \boldsymbol{\mu}_i{}^T\Sigma^{-1}x + \log\pi_i.$$

Канонические переменные

Задача: найти линейное преобразование $\mathbf{Z} = A^{\mathrm{T}}\mathbf{X}$, в результате которого получаются признаки наилучшим образом разделяющие группы.

Канонические переменные

Задача: найти линейное преобразование $\mathbf{Z} = A^{\mathrm{T}}\mathbf{X}$, в результате которого получаются признаки наилучшим образом разделяющие группы.
Вычислим внутриклассовую ковариационную матрицу:

$$\mathbf{E} = \frac{1}{n-K}\sum_{i=1}^{K}\sum_{j:y_j=Y_i}(x_j-\widehat{\mu}_i)^{\mathrm{T}}(x_j-\widehat{\mu}_i)$$

Вычисляем межклассовую ковариационную матрицу (с точностью до коэффициента):

$$\mathbf{H} = \sum_{i=1}^{K} n_i(\widehat{\mu}_i-\widehat{\mu})^{\mathrm{T}}(\widehat{\mu}_i-\widehat{\mu}).$$

Канонические переменные

Выборочная ковариационная матрица (с точностью до коэффициента) новых признаков имеет вид:

$$A^{\mathrm{T}}\mathbf{T}A = A^{\mathrm{T}}(\mathbf{E}+\mathbf{H})A = A^{\mathrm{T}}\mathbf{E}A + A^{\mathrm{T}}\mathbf{H}A,$$

где $\mathbf{T}$ – total covariance matrix, первое слагаемое – оценка внутригрупповых отклонений, а второе – оценка межгрупповых отклонений. Воспользовавшись критерием Фишера перейдем к обобщенной задаче на собственные числа и собственные вектора:

$$\frac{A^{\mathrm{T}}\mathbf{H}A}{A^{\mathrm{T}}\mathbf{E}A} \to \max_{A}.$$

Канонические переменные

Пусть $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$ – собственные числа матрицы $\mathbf{E}^{-1}\mathbf{H}$, а $A_1, \ldots, A_d$ – соответствующие им собственные вектора. Тогда максимум выше равен λ_1 и достигается на A_1. При этом $A_i^{\mathrm{T}}\mathbf{E}A_j = 0$. Далее

$$\max_{A, A \perp A_1} \frac{A^{\mathrm{T}}\mathbf{H}A}{A^{\mathrm{T}}\mathbf{E}A} = \lambda_2,$$

достигается на A_2 и так далее.
Вектора A_i называют каноническими коэффициентами, а новые признаки Z_i – каноническими переменными, Z_i ортогональны.

Значимость канонических переменных

Возникает вопрос: сколько канонических переменных нам окажется достаточно взять? Другими словами, нужно проверить гипотезу:

$$H_0 : A_i, i = \ell, \ldots, d \text{ не описывают отличия.}$$

- Wilks' Lambda
$$\Lambda_\ell^p = \prod_{i=l}^{d} \frac{1}{1+\lambda_i};$$
- Roy's greatest root
$$r_1^2 = \frac{\lambda_1}{1+\lambda_1};$$
- Pillai's trace
$$V = trace(\mathbf{H}(\mathbf{H}+\mathbf{E})^{-1});$$
- Hotelling-Lawley trace
$$V = trace(\mathbf{H}\mathbf{E}^{-1}).$$

QDA

Предполагаем, что каждый класс имеет многомерное нормальное распределение с различными ковариационными матрицами.

QDA

Предполагаем, что каждый класс имеет многомерное нормальное распределение с различными ковариационными матрицами.
Тогда плотность в точке x

$$p\left(x|\xi = Y_i\right) = \frac{1}{(2\pi)^{p/2}\left|\mathbf{\Sigma}_i\right|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T \mathbf{\Sigma}_i^{-1}(x-\mu_i)\right),$$

и классифицирующая функция $f_i\left(x\right) = \pi_i p\left(x|\xi = Y_i\right)$.
Применяем возрастающее монотонное преобразование и оставляем в классифицирующей функции только члены, отличающиеся в разных группах:

$$g_i\left(x\right) = \log f_i\left(x\right) = \log \pi_i - \frac{1}{2}\log\left|\mathbf{\Sigma}_i\right| - \frac{1}{2}(x-\mu_i)^T \mathbf{\Sigma}_i^{-1}(x-\mu_i),$$

получаем квадратично зависящую от x классифицирующую функцию.

Наивный байесовский классификатор

Предположим, что признаки независимы внутри групп и имеют нормальное распределение:

Наивный байесовский классификатор

Предположим, что признаки независимы внутри групп и имеют нормальное распределение:

$$p_i(x) = \prod_{j=1}^{p} p_{ij}(x_j), \quad p_{ij}(x_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{(x_j-\mu_{ij})^2}{2\sigma_{ij}^2}}.$$

Отсюда классифицирующую функцию можно представить в виде:

$$\delta_i(x) = -\frac{1}{2}\sum_{j=1}^{p} \frac{(x_j-\mu_{ij})^2}{2\sigma_{ij}^2} + \log(\pi_i).$$

Cross-validation

- Имеется выборка $(X,\ Y)$;
- Строим модель, зависящую от параметра θ и минимизирующую ошибку $J(X, Y; \theta, \lambda)$, где λ — параметр регуляризации.

Cross-validation

- Имеется выборка $(X,\ Y)$;
- Строим модель, зависящую от параметра θ и минимизирующую ошибку $J(X, Y; \theta, \lambda)$, где λ — параметр регуляризации.

Хотим подобрать такой параметр $\hat{\theta}$, чтобы минимизировать ошибку $J(X_{new}, Y_{new}; \hat{\theta}, 0)$ на новых индивидах.
Вариант решения:

- Делим выборку $(X,\ Y)$ случайным образом на три набора: $(X_{train},\ Y_{train})$, $(X_{CV},\ Y_{CV})$ и $(X_{test},\ Y_{test})$;
- Перебираем набор параметров$\lambda_1, \ldots, \lambda_m$;
- Для каждого параметра λ_i строим модель на $(X_{train},\ Y_{train})$ и считаем ошибку $J(X_{CV}, Y_{CV}; \theta_i, 0)$;
- Берем $\lambda = \lambda_0$ с минимальной ошибкой (ему соответствует $\hat{\theta}$);
- Считаем ошибку модели $J(X_{test}, Y_{test}; \hat{\theta}, \lambda_0)$.

K-fold Cross-validation

Алгоритм:

- Делим выборку $(X,\ Y)$ случайным образом на K частей: $(X_1,\ Y_1)$,...,$(X_K,\ Y_K)$;
- Обозначим $(X_k',\ Y_k')$ набор, содержащий всех индивидов, кроме $(X_k,\ Y_k)$;
- Перебираем набор параметров $\lambda_1, \ldots, \lambda_m$;
- Для каждого параметра λ_i считаем:

$$CV_i = \sum_{j=1}^{K} \frac{n_j}{n} J(X_j, Y_j; \theta_j, 0),$$

где θ_j минимизирует $J(X_j', Y_j'; \theta, \lambda_i)$, n_j — число индивидов в $(X_j,\ Y_j)$;

- Берем $\lambda = \lambda_0$ с минимальной ошибкой CV_i;
- Берем $\hat{\theta}$, которое минимизирует $J(X, Y; \theta, \lambda_0)$.