ВВЕДЕНИЕ В МАШИННОЕ ОБУЧЕНИЕ

Лекция №4

План лекции

1. Support Vector Machine

2. Principal component analysis

01

SVM

Линейная классификация

Вспомним, что такое модель линейной классификации

$$X^l = (x_i, y_i)_{i=1}^l, x_i \in R^n, y_i \in \{-1, 1\}$$

$$a(x; w, w_0) = \mathrm{sign}\big(\langle x, w\rangle - w_0\big) \qquad w \in R^n, w_0 \in R$$

Где функция ошибки:

$$\sum_{i=1}^{\ell}\big[a(x_i; w, w_0) \neq y_i\big] = \sum_{i=1}^{\ell}\big[M_i(w, w_0) < 0\big] \to \min_{w, w_0}.$$

$$M_i(w, w_0) = \big(\langle x_i, w\rangle - w_0\big)y_i$$

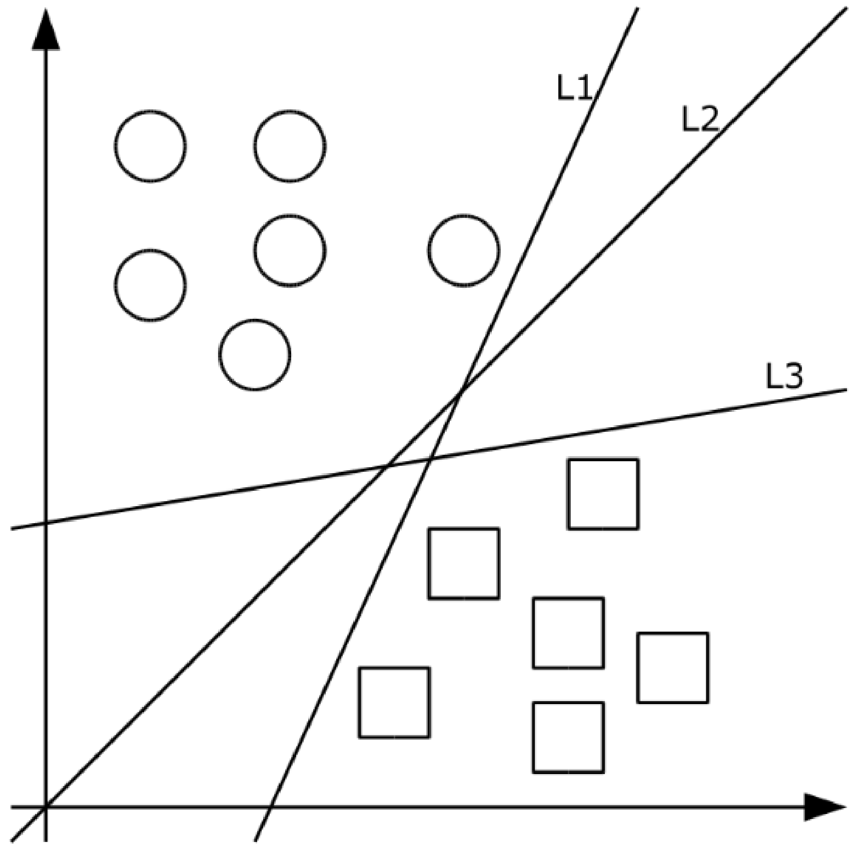
Решение классификации

Вспомним, что такое модель линейной классификации

$$X^l = (x_i, y_i)_{i=1}^l, x_i \in R^n, y_i \in \{-1, 1\}$$

$$\exists w, w_0 : \quad M_i(w, w_0) = y_i\big(\langle w, x_i \rangle - w_0\big) > 0, \quad i = 1, \dots, \ell$$

Какое решение l1, l2, l3 лучше всего в данном случае?



Решение классификации

Вспомним, что такое модель линейной классификации

$$X^l = (x_i, y_i)_{i=1}^l, x_i \in R^n, y_i \in \{-1, 1\}$$

$$\exists w, w_0 : \quad M_i(w, w_0) = y_i(\langle w, x_i \rangle - w_0) > 0, \quad i = 1, \dots, \ell$$

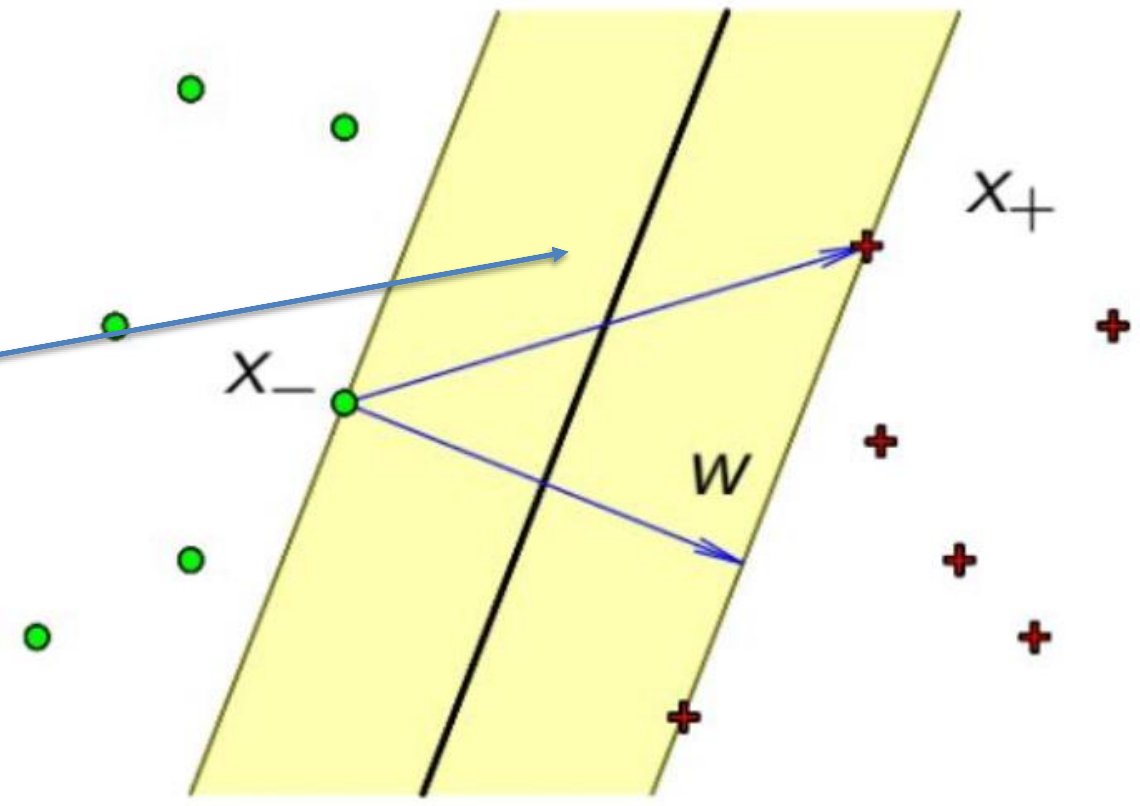
Введем такую нормировку: $\min\limits_{i=1,\dots,\ell} M_i(w, w_0) = 1$

Теперь отступ между самыми близкими объектами классов равен 1

$$\{x : -1 \le (< w, x > -w_0) \le 1\}$$

$$\forall x_+ : \ \langle w, x_+ \rangle \quad -w_0 \ge 1$$

$$\forall x_- : \ \langle w, x_- \rangle \quad -w_0 \le -1$$



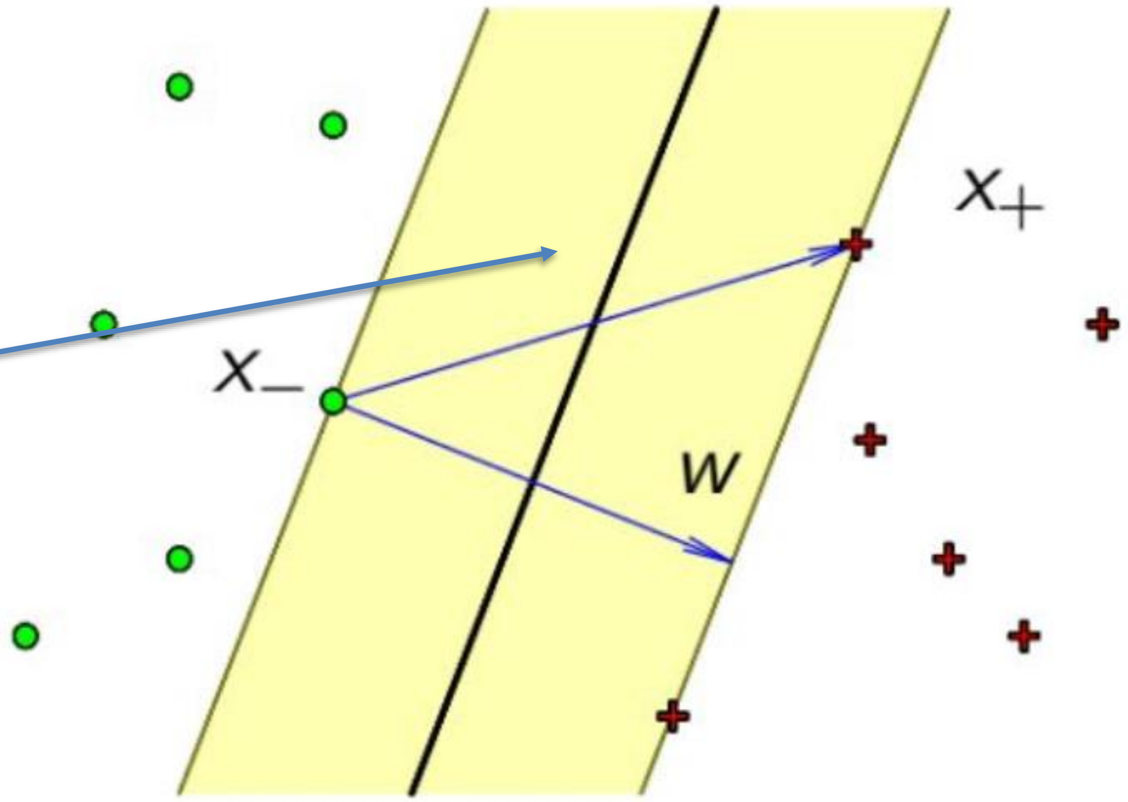
Решение классификации

Теперь отступ между самыми близкими объектами классов равен 1

$$\{x : -1 \le (< w, x > -w_0) \le 1\}$$

$$\forall x_+ : \langle w, x_+ \rangle \quad -w_0 \ge 1$$

$$\forall x_- : \langle w, x_- \rangle \quad -w_0 \le -1$$



Поставим задачу максимизации такого отступа

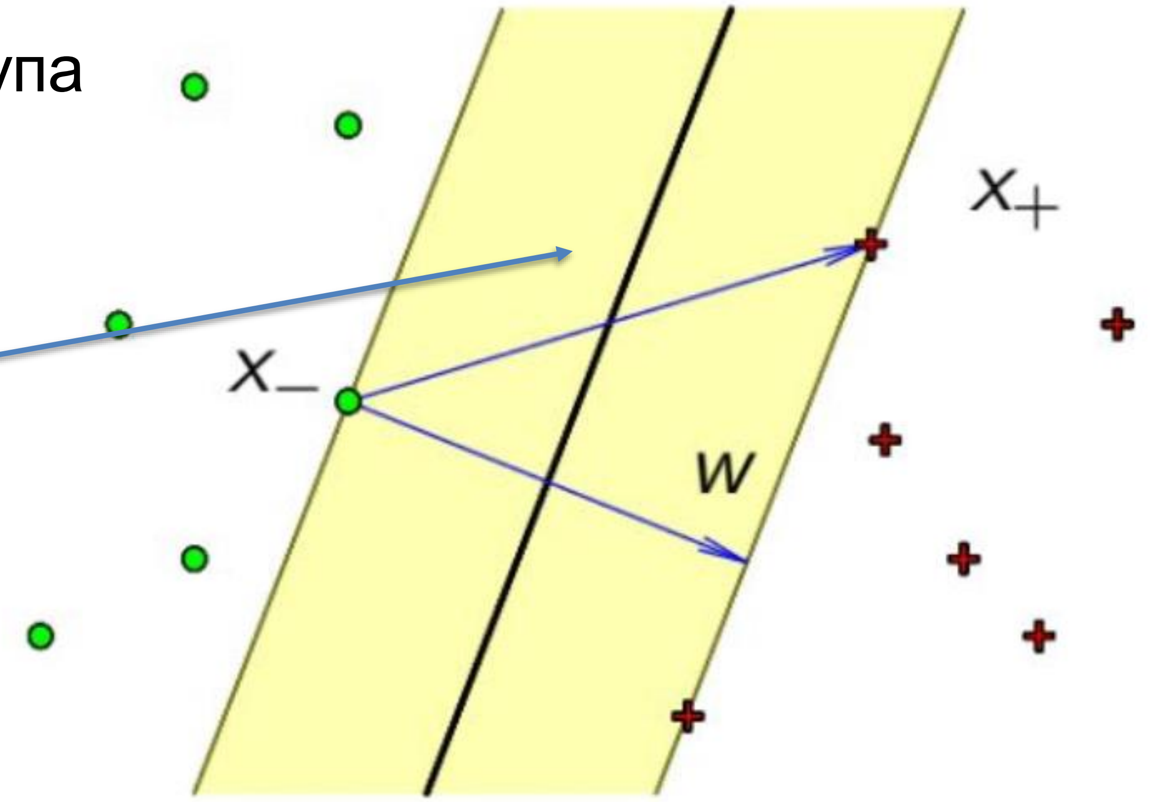
$$\frac{\langle\, x_+ - x_-, w \,\rangle}{||w||} \ge \boxed{\frac{2}{||w||}} \to max$$

За счет введенной нормировки M

Идеальный случай

Поставим задачу максимизации такого отступа

$$\begin{cases} \frac{1}{2}\|w\|^2 \to \min\limits_{w,w_0}; \\ M_i(w, w_0) \geqslant 1, \quad i = 1, \dots, \ell. \end{cases}$$



Нелинейная классификация

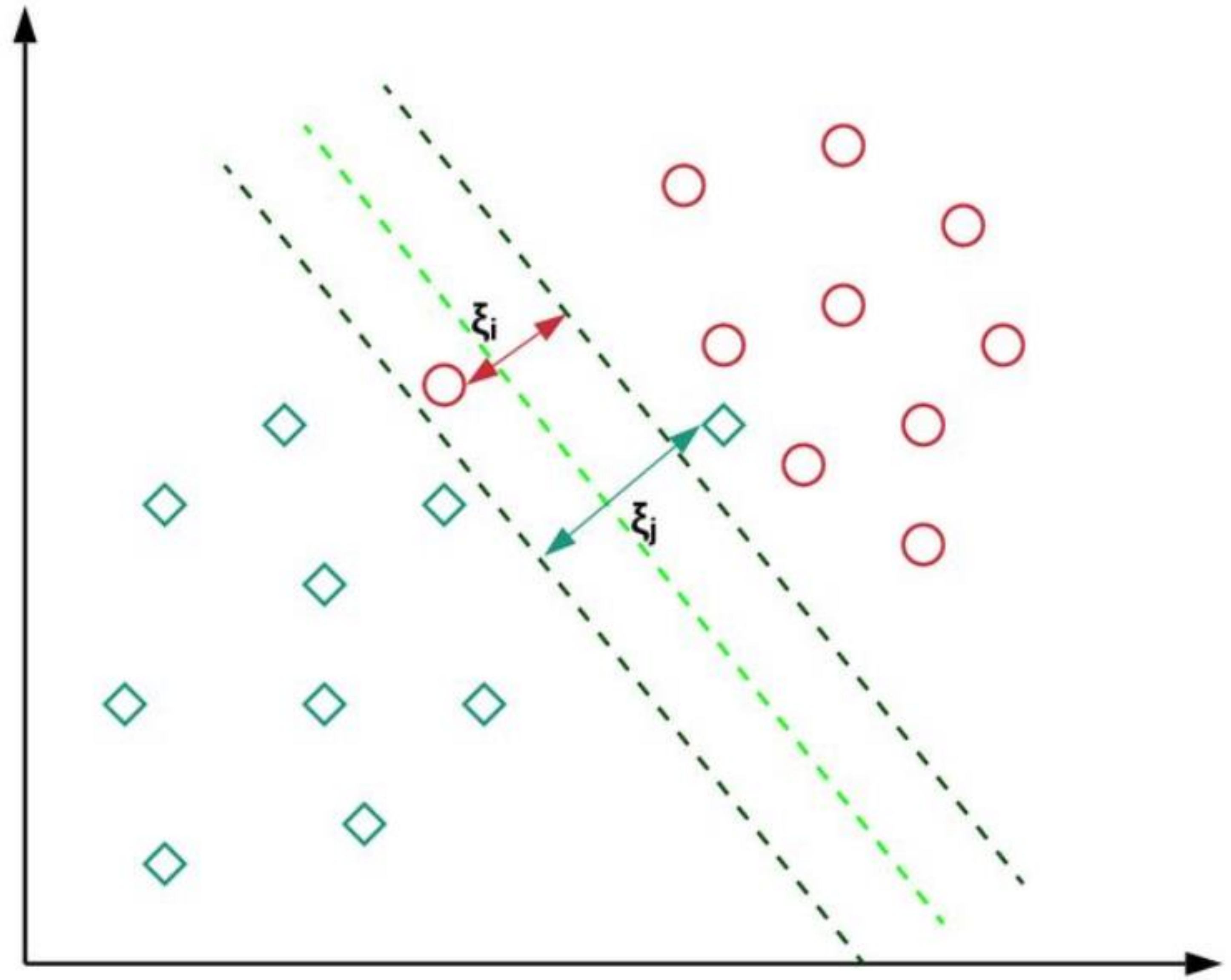
Поставим задачу максимизации такого отступа

$$\begin{cases} \frac{1}{2}\|w\|^2 \to \min\limits_{w,w_0}; \\ M_i(w, w_0) \geqslant 1, \quad i = 1, \dots, \ell. \end{cases}$$

Введем эмпирику

$$M_i(w, w_0) \geqslant 1 \boxed{- \xi_i,} \quad i = 1, \dots, \ell;$$

Допускаем, что классификатор ошибается



Нелинейная классификация

Оптимизация с ограничениями

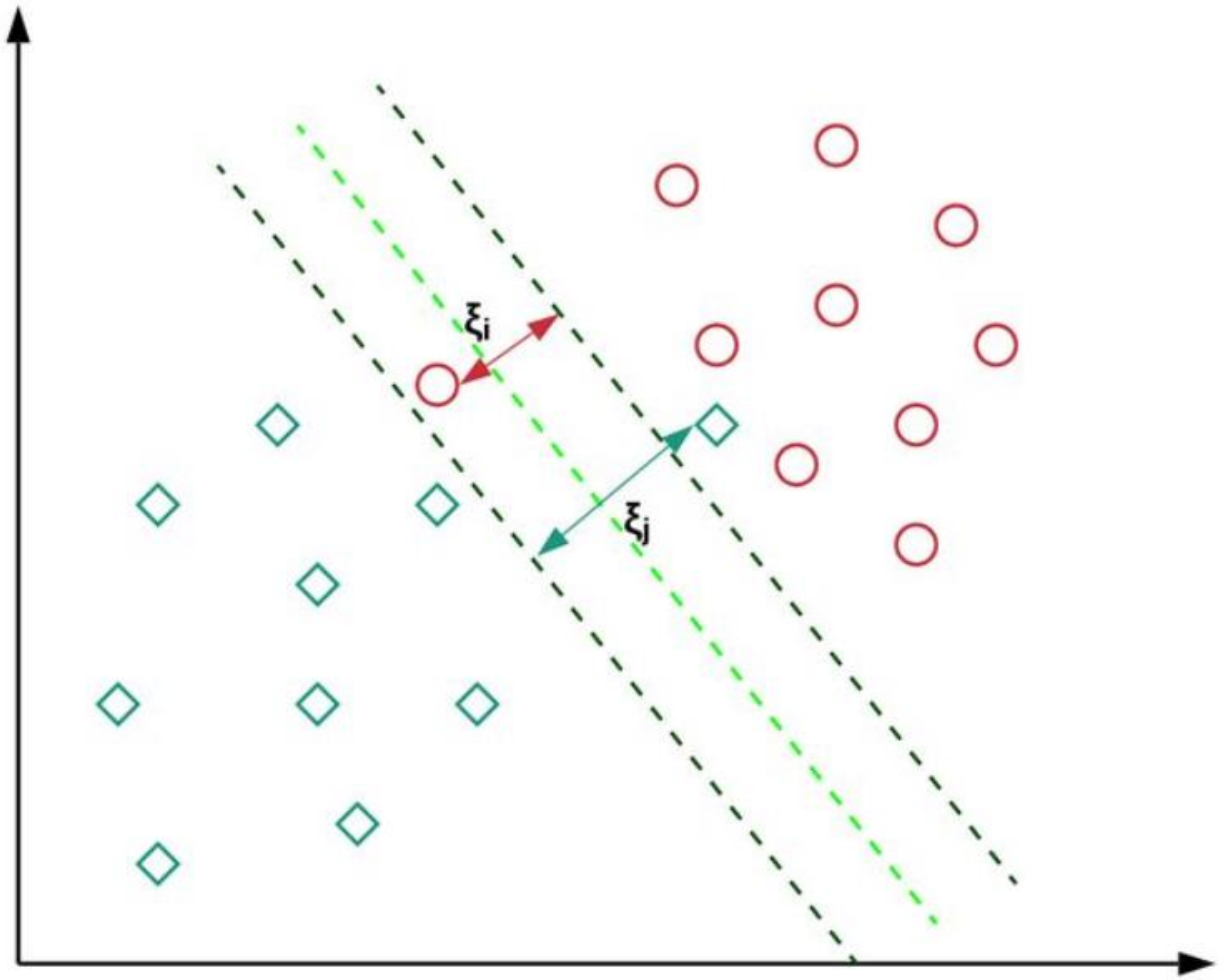
$$\begin{cases} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}\xi_i \to \min\limits_{w,w_0,\xi}; \\ M_i(w,w_0) \geqslant 1-\xi_i, \quad i=1,\dots,\ell; \\ \xi_i \geqslant 0, \quad i=1,\dots,\ell. \end{cases}$$

Эквивалентная задача безусловной оптимизации

$$\boxed{C\sum_{i=1}^{\ell}\left(1-M_i(w,w_0)\right)_+} + \frac{1}{2}\|w\|^2 \to \min_{w,w_0}.$$

Кусочно-линейная

С – скалярная величина, сила регуляризации (гиперпараметр)



Геометрическая постановка

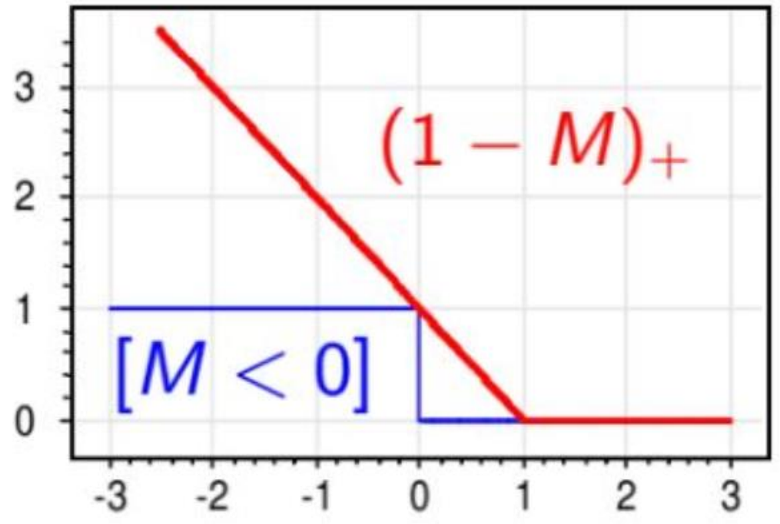
[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Регуляризация

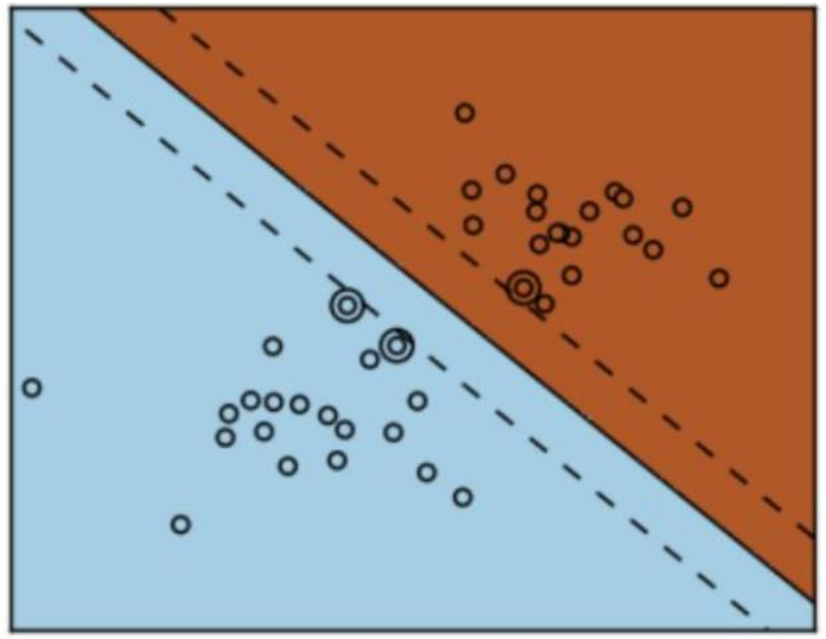
Оптимизация с ограничениями

$$Q(w, w_0) = \sum_{i=1}^{l}[M_i(w, w_0) < 0] \leq$$

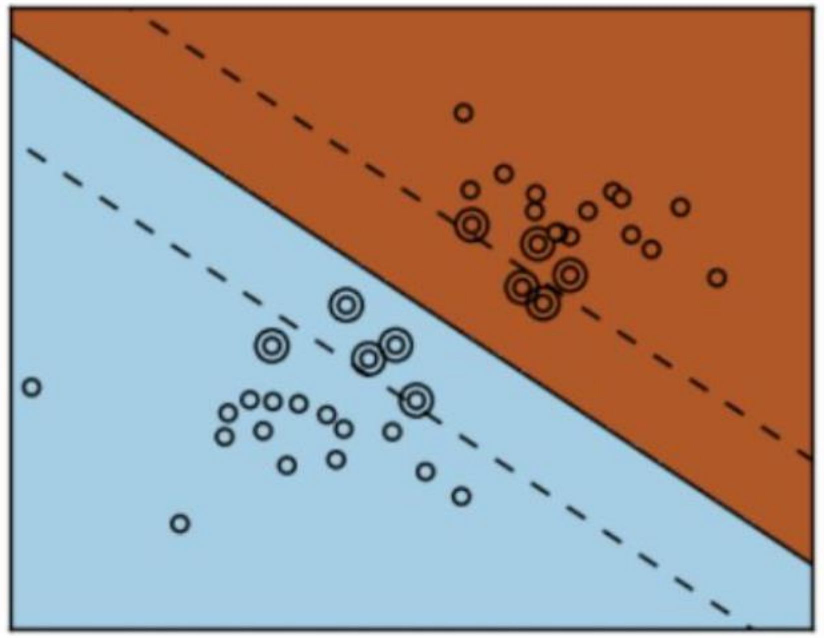
$$\leq \underbrace{\sum_{i=1}^{l}(1 - M_i(w, w_0))_+}_{\text{Approximation}} + \underbrace{\frac{1}{2C}||w||^2}_{\text{Regularization}} \to min$$



С большая: слабая оптимизация



С маленькая: сильная оптимизация



Нелинейная классификация

Задача квадратичного программирования. Теорема Каруша-Куна-Таккера

$$\begin{cases} f(x) \to \min\limits_{x}; \\ g_i(x) \leqslant 0, \quad i = 1, \ldots, m; \\ h_j(x) = 0, \quad j = 1, \ldots, k. \end{cases}$$

Необходимые условия. Если x — точка локального минимума, то существуют множители μ_i, $i = 1, \ldots, m$, λ_j, $j = 1, \ldots, k$:

$$\begin{cases} \dfrac{\partial \mathscr{L}}{\partial x} = 0, \quad \mathscr{L}(x; \mu, \lambda) = f(x) + \sum\limits_{i=1}^{m} \mu_i g_i(x) + \sum\limits_{j=1}^{k} \lambda_j h_j(x); \\ g_i(x) \leqslant 0; \ h_j(x) = 0; \ \text{(исходные ограничения)} \\ \mu_i \geqslant 0; \ \text{(двойственные ограничения)} \\ \mu_i g_i(x) = 0; \ \text{(условие дополняющей нежёсткости)} \end{cases}$$

[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]()

Нелинейная классификация

Задача квадратичного программирования. Теорема Каруша-Куна-Таккера

Функция Лагранжа: $\mathscr{L}(w, w_0, \xi; \lambda, \eta) =$

$$= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \lambda_i \big(M_i(w, w_0) - 1\big) - \sum_{i=1}^{\ell} \xi_i (\lambda_i + \eta_i - C),$$

λ_i — переменные, двойственные к ограничениям $M_i \geqslant 1 - \xi_i$;
η_i — переменные, двойственные к ограничениям $\xi_i \geqslant 0$.

$$\begin{cases} \frac{\partial \mathscr{L}}{\partial w} = 0, \quad \frac{\partial \mathscr{L}}{\partial w_0} = 0, \quad \frac{\partial \mathscr{L}}{\partial \xi} = 0; \\ \xi_i \geqslant 0, \quad \lambda_i \geqslant 0, \quad \eta_i \geqslant 0, \quad i = 1, \dots, \ell; \\ \lambda_i = 0 \ \text{либо} \ M_i(w, w_0) = 1 - \xi_i, \quad i = 1, \dots, \ell; \\ \eta_i = 0 \ \text{либо} \ \xi_i = 0, \quad i = 1, \dots, \ell; \end{cases}$$

[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Нелинейная классификация

Задача квадратичного программирования. Теорема Каруша-Куна-Таккера

Функция Лагранжа: $\mathscr{L}(w, w_0, \xi; \lambda, \eta) =$

$$= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \lambda_i \big(M_i(w, w_0) - 1\big) - \sum_{i=1}^{\ell} \xi_i (\lambda_i + \eta_i - C),$$

Необходимые условия седловой точки функции Лагранжа:

$$\frac{\partial \mathscr{L}}{\partial w} = w - \sum_{i=1}^{\ell} \lambda_i y_i x_i = 0 \quad \Longrightarrow \quad w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\frac{\partial \mathscr{L}}{\partial w_0} = -\sum_{i=1}^{\ell} \lambda_i y_i = 0 \quad \Longrightarrow \quad \sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 \quad \Longrightarrow \quad \eta_i + \lambda_i = C, \quad i = 1, \ldots, \ell.$$

[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Понятия опорного вектора

Типизация объектов:

1. $\lambda_i = 0;\ \ \eta_i = C;\ \ \xi_i = 0;\ \ M_i \geqslant 1.$
   — периферийные (неинформативные) объекты.
2. $0 < \lambda_i < C;\ \ 0 < \eta_i < C;\ \ \xi_i = 0;\ \ M_i = 1.$
   — опорные граничные объекты.
3. $\lambda_i = C;\ \ \eta_i = 0;\ \ \xi_i > 0;\ \ M_i < 1.$
   — опорные-нарушители.

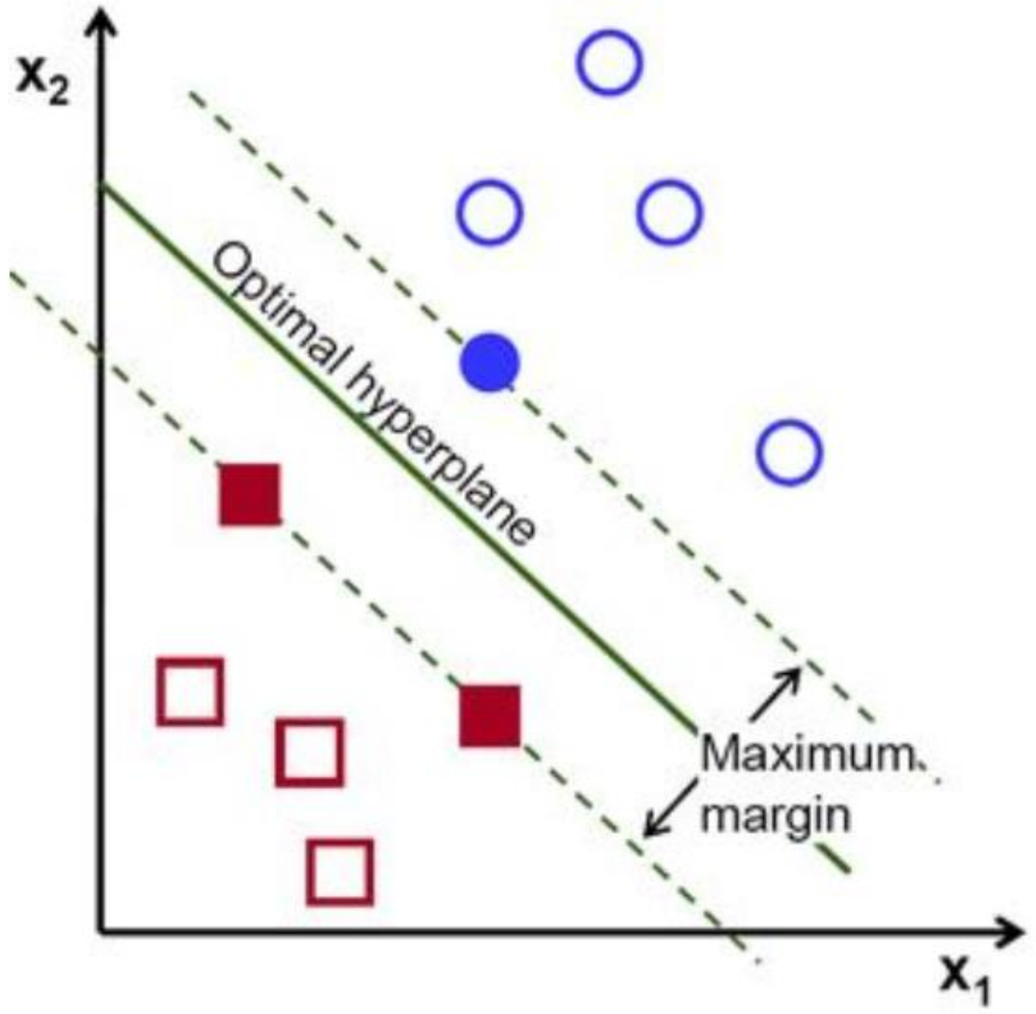
Определение

Объект x_i называется *опорным*, если $\lambda_i \neq 0$.

$$w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\eta_i + \lambda_i = C, \quad i = 1, \dots, \ell.$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Понятия опорного вектора

Решение не зависит от элемента выборки i

Класс находится за оптимальной гиперплоскостью класса. Нет ошибки
От него не зависит решение

Типизация объектов:

1. $\boxed{\lambda_i = 0;}$ $\eta_i = C;$ $\xi_i = 0;$ $\boxed{M_i \geqslant 1.}$
   — периферийные (неинформативные) объекты.
2. $0 < \lambda_i < C;$ $0 < \eta_i < C;$ $\xi_i = 0;$ $M_i = 1.$
   — опорные граничные объекты.
3. $\lambda_i = C;$ $\eta_i = 0;$ $\xi_i > 0;$ $M_i < 1.$
   — опорные-нарушители.

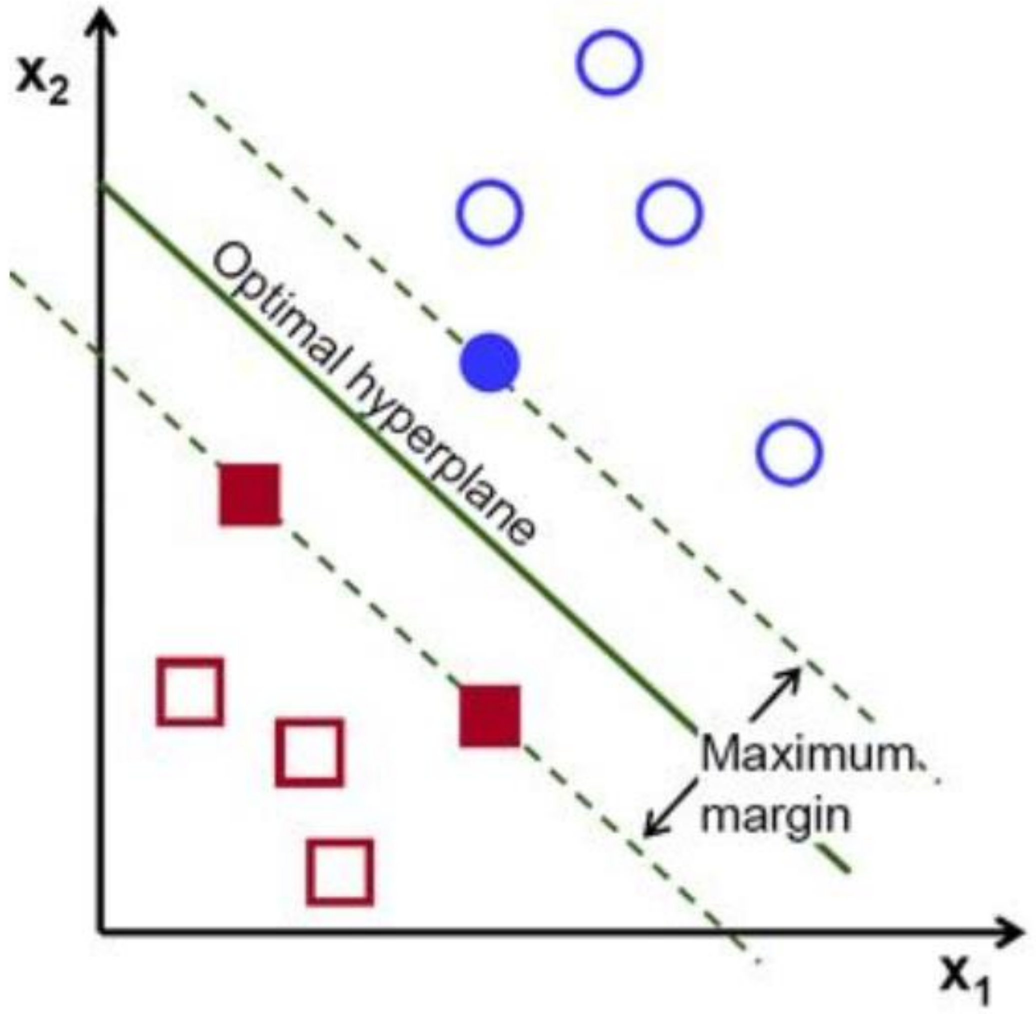
Определение

Объект x_i называется *опорным*, если $\lambda_i \neq 0$.

$$w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\eta_i + \lambda_i = C, \quad i = 1, \dots, \ell.$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Понятия опорного вектора

$$w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\eta_i + \lambda_i = C, \quad i = 1, \dots, \ell.$$

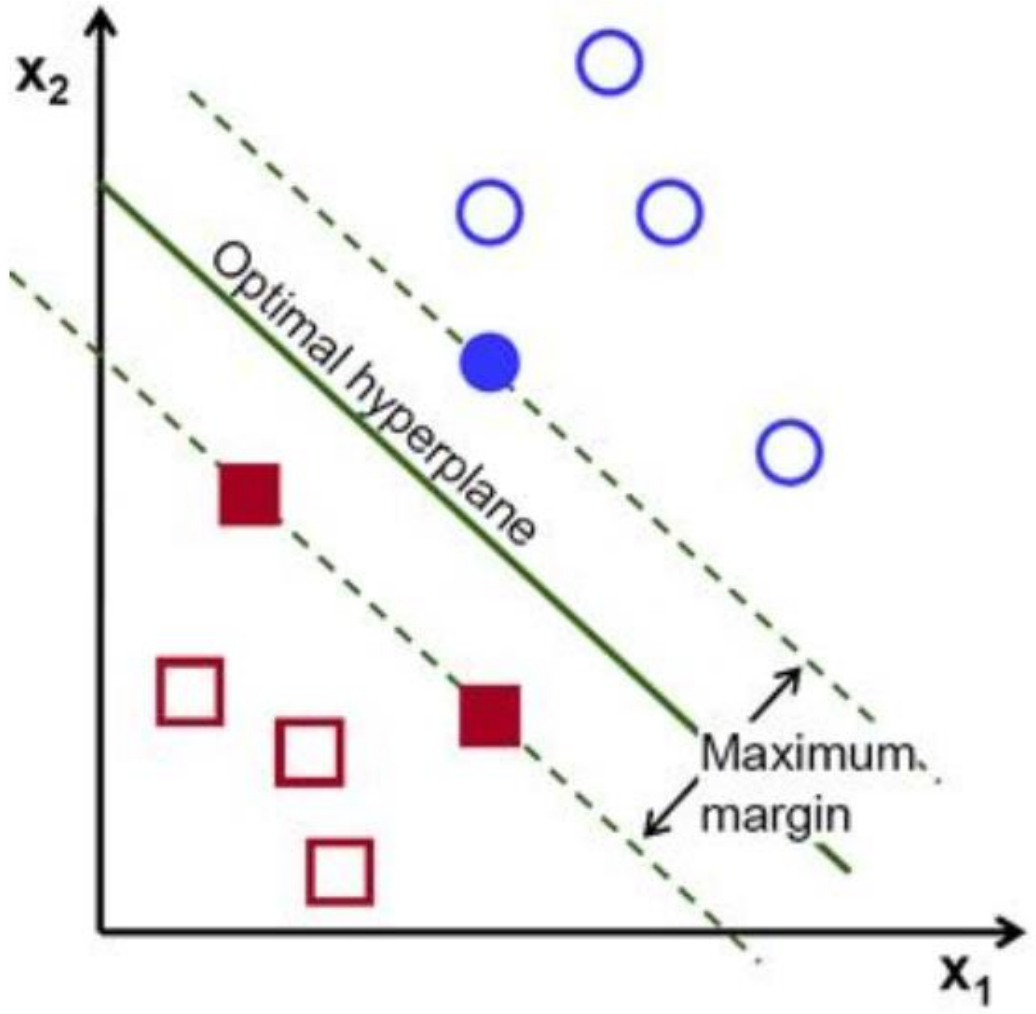
Типизация объектов:

1. $\lambda_i = 0;\ \ \eta_i = C;\ \ \xi_i = 0;\ \ M_i \geqslant 1.$
   — периферийные (неинформативные) объекты.
2. $0 < \lambda_i < C;\ \ 0 < \eta_i < C;\ \ \boxed{\xi_i = 0;\ \ M_i = 1.}$
   — опорные граничные объекты.
3. $\lambda_i = C;\ \ \eta_i = 0;\ \ \xi_i > 0;\ \ M_i < 1.$
   — опорные-нарушители.

Лежат ровно на границе

Определение

Объект x_i называется *опорным*, если $\lambda_i \neq 0$.



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Понятия опорного вектора

Типизация объектов:

1. $\lambda_i = 0;\ \ \eta_i = C;\ \ \xi_i = 0;\ \ M_i \geqslant 1.$
   — периферийные (неинформативные) объекты.
2. $0 < \lambda_i < C;\ \ 0 < \eta_i < C;\ \ \xi_i = 0;\ \ M_i = 1.$
   — опорные граничные объекты.
3. $\lambda_i = C;\ \ \eta_i = 0;\ \boxed{\xi_i > 0;\ \ M_i < 1.}$
   — опорные-нарушители.

Могут сидеть в «чужом» классе

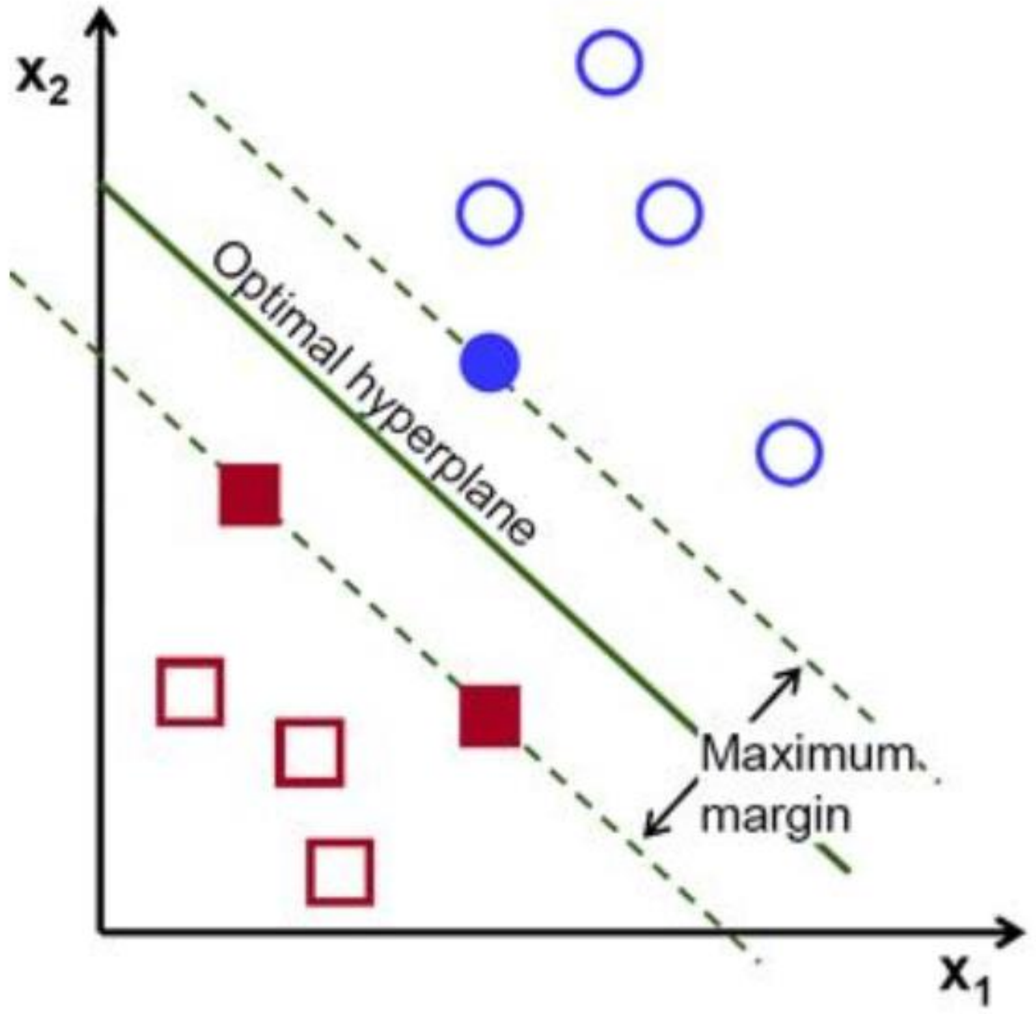
Определение

Объект x_i называется *опорным*, если $\lambda_i \neq 0$.

$$w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\eta_i + \lambda_i = C, \quad i = 1, \dots, \ell.$$



Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс

Двойственная задача

Подставим все ограничения через w и перепишем Лагранжиан через λ_i

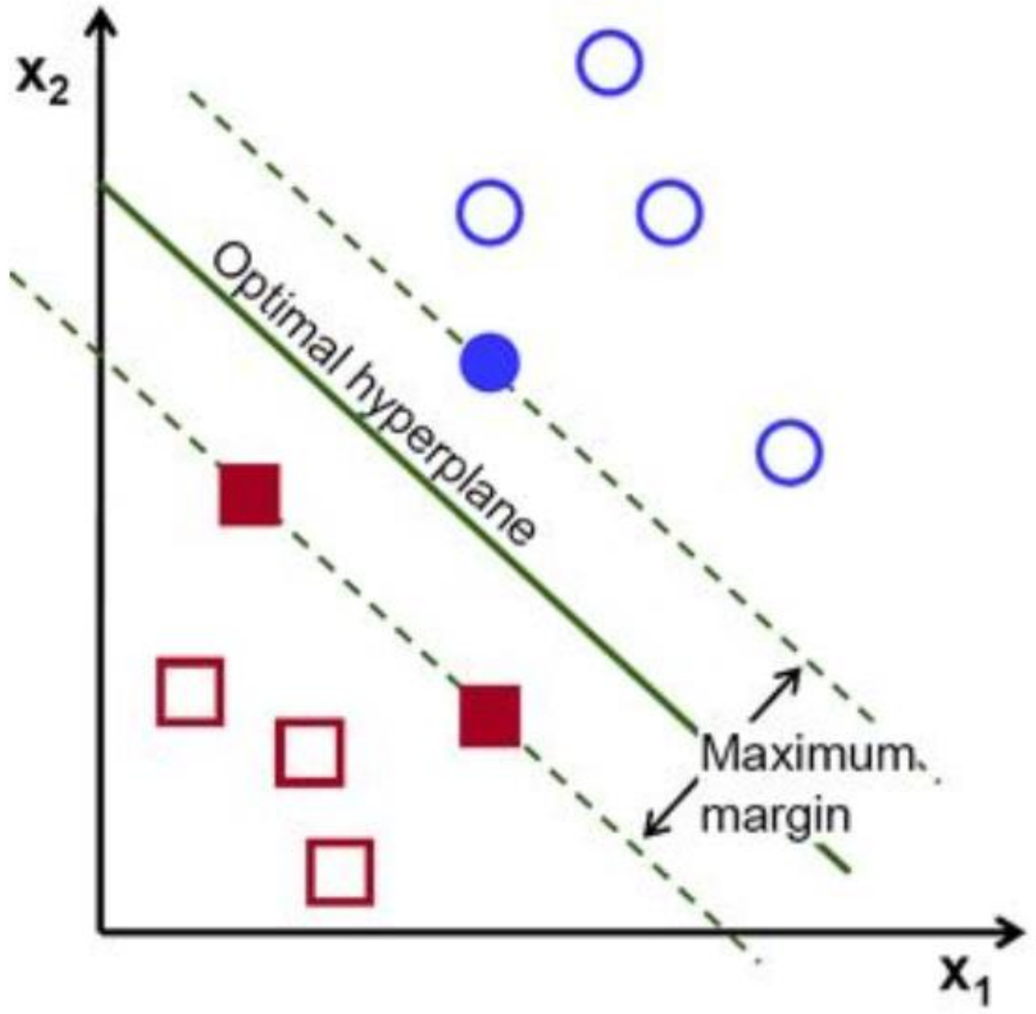
$$\begin{cases} -\mathscr{L}(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle \to \min\limits_{\lambda}; \\ 0 \leqslant \lambda_i \leqslant C, \quad i = 1, \dots, \ell; \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0. \end{cases}$$

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum_{i=1}^{\ell} \lambda_i y_i x_i; \\ w_0 = \langle w, x_i \rangle - y_i, \quad \text{для любого } i\text{: } \lambda_i > 0, \ M_i = 1. \end{cases}$$

Линейный классификатор с признаками $f_i(x) = \langle x, x_i \rangle$:

$$a(x) = \operatorname{sign}\left(\sum_{i=1}^{\ell} \lambda_i y_i \langle x, x_i \rangle - w_0 \right).$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Двойственная задача

Подставим все ограничения через w и перепишем Лагранжиан через λ_i

Матрица Грамма (скалярное произведение)

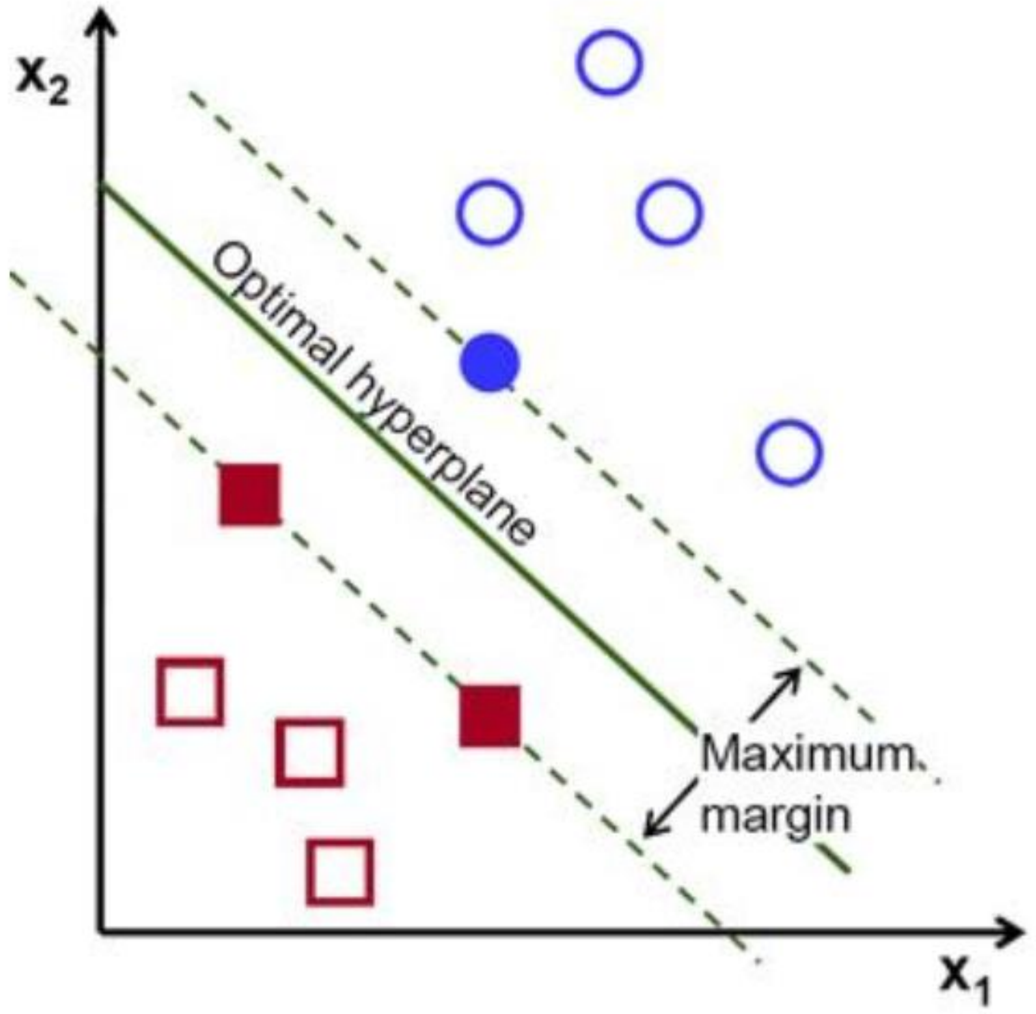
$$\begin{cases} -\mathscr{L}(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \boxed{\langle x_i, x_j \rangle} \to \min\limits_{\lambda}; \\ 0 \leqslant \lambda_i \leqslant C, \quad i = 1, \dots, \ell; \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0. \end{cases}$$

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum_{i=1}^{\ell} \lambda_i y_i x_i; \\ w_0 = \langle w, x_i \rangle - y_i, \quad \text{для любого } i\colon \lambda_i > 0,\ M_i = 1. \end{cases}$$

Линейный классификатор с признаками $f_i(x) = \langle x, x_i \rangle$:

$$a(x) = \operatorname{sign}\Big(\sum_{i=1}^{\ell} \lambda_i y_i \langle x, x_i \rangle - w_0 \Big).$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Двойственная задача

Подставим все ограничения через w и перепишем Лагранжиан через λ_i

Матрица Грамма (скалярное произведение)

$$\begin{cases} -\mathscr{L}(\lambda) = -\sum\limits_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum\limits_{i=1}^{\ell} \sum\limits_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \boxed{\langle x_i, x_j \rangle} \to \min\limits_{\lambda}; \\ \boxed{0 \leqslant \lambda_i \leqslant C, \quad i = 1}, \dots, \ell; \\ \sum\limits_{i=1}^{\ell} \lambda_i y_i = 0. \end{cases}$$

Куб с ребром С

Выпуклая задача

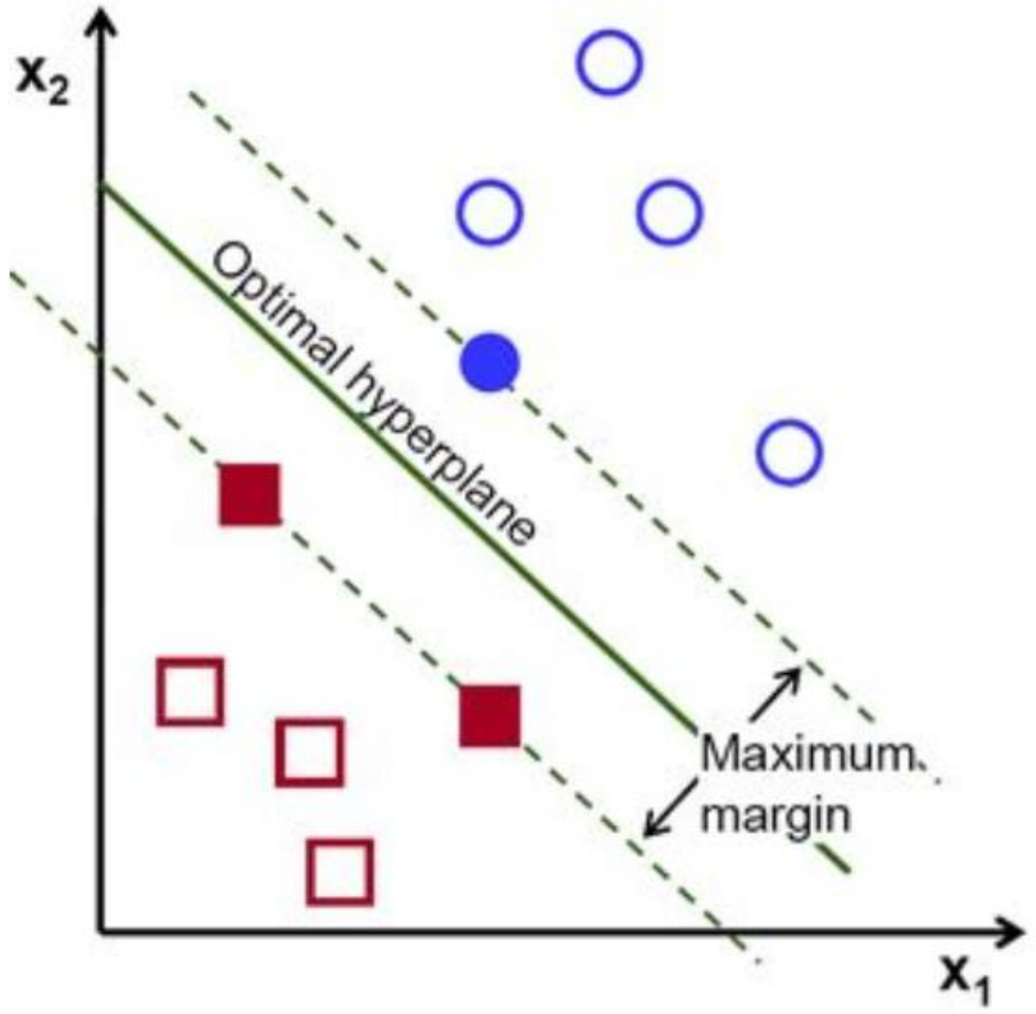
Гиперплоскость

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum\limits_{i=1}^{\ell} \lambda_i y_i x_i; \\ w_0 = \langle w, x_i \rangle - y_i, \quad \text{для любого } i\text{: } \lambda_i > 0,\ M_i = 1. \end{cases}$$

Линейный классификатор с признаками $f_i(x) = \langle x, x_i \rangle$:

$$a(x) = \operatorname{sign}\left(\sum_{i=1}^{\ell} \lambda_i y_i \langle x, x_i \rangle - w_0 \right).$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Двойственная задача

Подставим все ограничения через w и перепишем Лагранжиан через λ_i

Матрица Грамма (скалярное произведение)

$$\begin{cases} -\mathscr{L}(\lambda) = -\sum_{i=1}^{\ell}\lambda_i + \frac{1}{2}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell}\lambda_i\lambda_j y_i y_j \boxed{\langle x_i, x_j\rangle} \to \min_{\lambda}; \\ \boxed{0 \leqslant \lambda_i \leqslant C, \quad i = 1}, \dots, \ell; \\ \sum_{i=1}^{\ell}\lambda_i y_i = 0. \end{cases}$$

Куб с ребром С

Выпуклая задача

Гиперплоскость

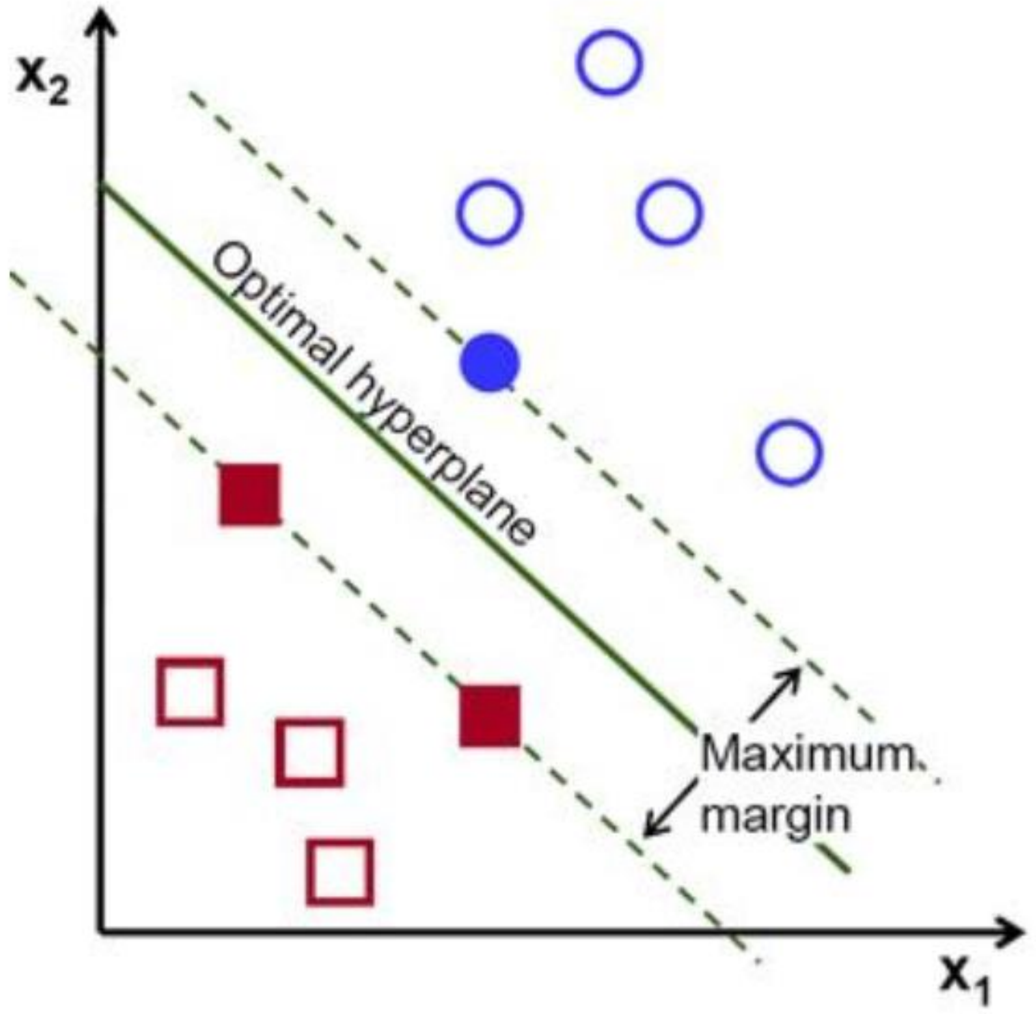
Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum_{i=1}^{\ell}\lambda_i y_i x_i; \\ w_0 = \langle w, x_i\rangle - y_i, \quad \text{для любого } i\colon \lambda_i > 0,\ M_i = 1. \end{cases}$$

На практике опорных векторов не так много

Линейный классификатор с признаками $f_i(x) = \langle x, x_i\rangle$:

$$a(x) = \operatorname{sign}\Big(\sum_{i=1}^{\ell}\lambda_i y_i\langle x, x_i\rangle - w_0\Big).$$



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

Двойственная задача

Подставим все ограничения через w и перепишем Лагранжиан через λ_i

$$\begin{cases} -\mathscr{L}(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \boxed{\langle x_i, x_j \rangle} \to \min_{\lambda}; \\ \boxed{0 \leqslant \lambda_i \leqslant C, \quad i = 1}, \dots, \ell; \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0. \end{cases}$$

Матрица Грамма (скалярное произведение)

Куб с ребром С

Выпуклая задача

Гиперплоскость

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum_{i=1}^{\ell} \lambda_i y_i x_i; \\ w_0 = \langle w, x_i \rangle - y_i, \quad \text{для любого } i\text{: } \lambda_i > 0, \ M_i = 1. \end{cases}$$

На практике опорных векторов не так много

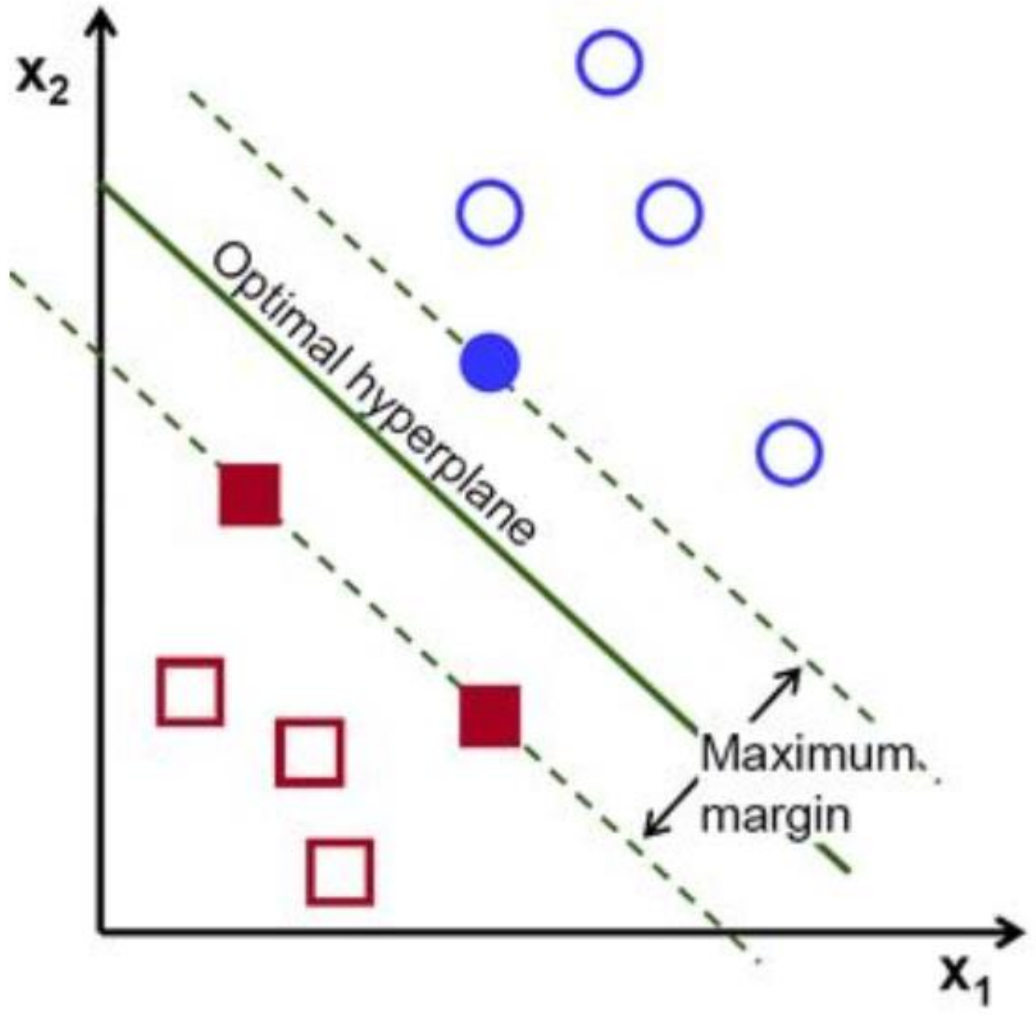
Линейный классификатор с признаками $f_i(x) = \langle x, x_i \rangle$:

Задача линейна относительно λ_i

$$a(x) = \mathrm{sign}\Big(\sum_{i=1}^{\ell} \lambda_i y_i \boxed{\langle x, x_i \rangle} - w_0 \Big).$$

Близость объекта до «опорного»

[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]



Трюк ядра

Идея: заменить $\langle x, x' \rangle$ нелинейной функцией $K(x, x')$.

Переход к спрямляющему пространству,
как правило, более высокой размерности: $\psi: X \to H$.

Определение

Функция $K: X \times X \to \mathbb{R}$ — *ядро*, если $K(x, x') = \langle \psi(x), \psi(x') \rangle$ при некотором $\psi: X \to H$, где H — гильбертово пространство.

Теорема

Функция $K(x, x')$ является ядром тогда и только тогда, когда

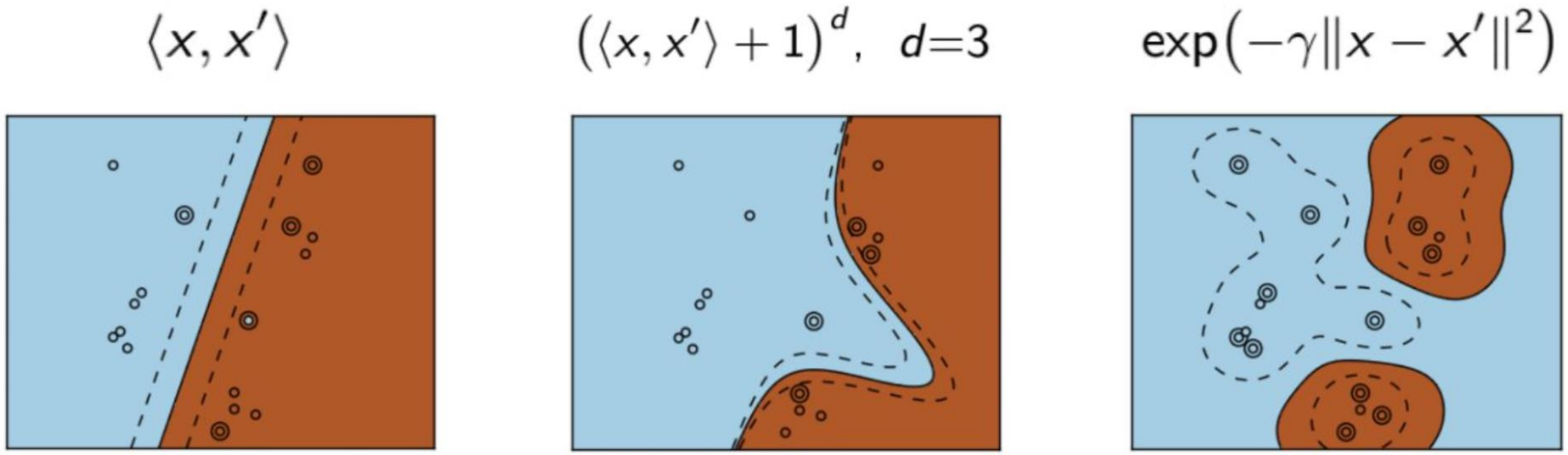
- она симметрична: $K(x, x') = K(x', x)$;
- и неотрицательно определена:
  $\int_X \int_X K(x, x') g(x) g(x') dx dx' \geqslant 0$ для любой $g: X \to \mathbb{R}$.

Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс

Трюк ядра

$$a(x) = \text{sign}\Big(\sum_{i=1}^{\ell} \lambda_i y_i \langle x, x_i \rangle - w_0\Big).$$

$K(x, x') =< x, x' >^2$ - quadratic
$K(x, x') =< x, x' >^d$ - polynomial with degree d
$K(x, x') = (< x, x' > +1)^d$ - polynomial with degree $\leq d$
$K(x, x') = exp(-\gamma||x - x'||^2)$ - Radial Basis Functions (RBF) kernel



[Машинное обучение. Метод опорных векторов. К.В. Воронцов, Школа анализа данных, Яндекс]

02

PCA

Задача понижения размерности

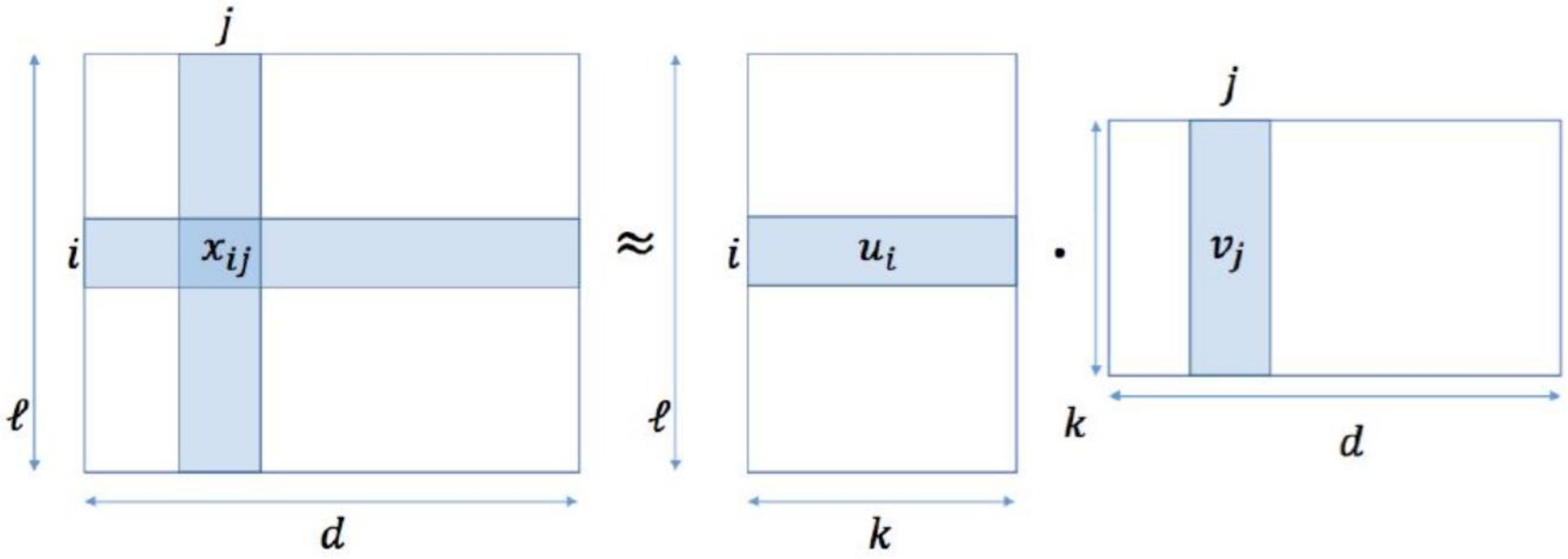
- В ML мы часто работаем с многомерными данными.
- Сотни или тысячи функций

- Их трудно визуализировать
- Обучение занимает много времени
- Некоторые модели хуже справляются с многомерными разреженными входными данными

Композиция матриц

Разложение на множители матриц меньшего ранга

$$X_{l,d} \approx U_{l,k} \cdot V_{k,d}^T \qquad ||X - UV^T|| \to min$$

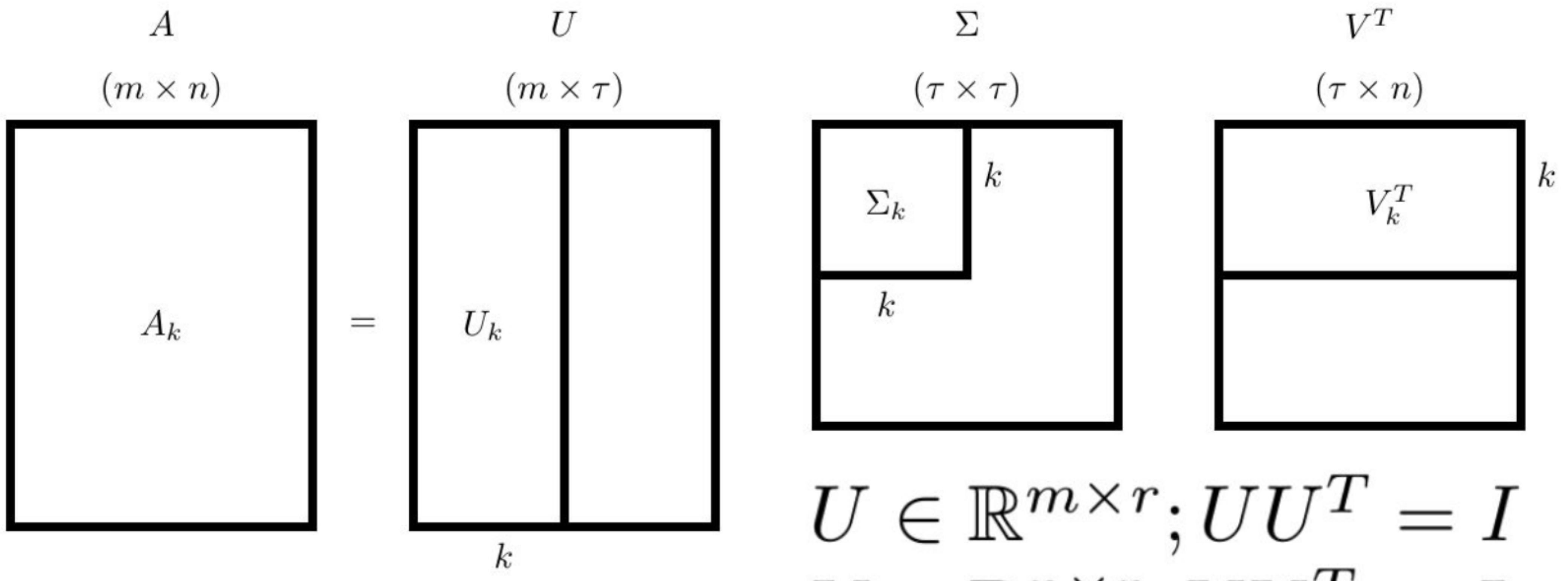


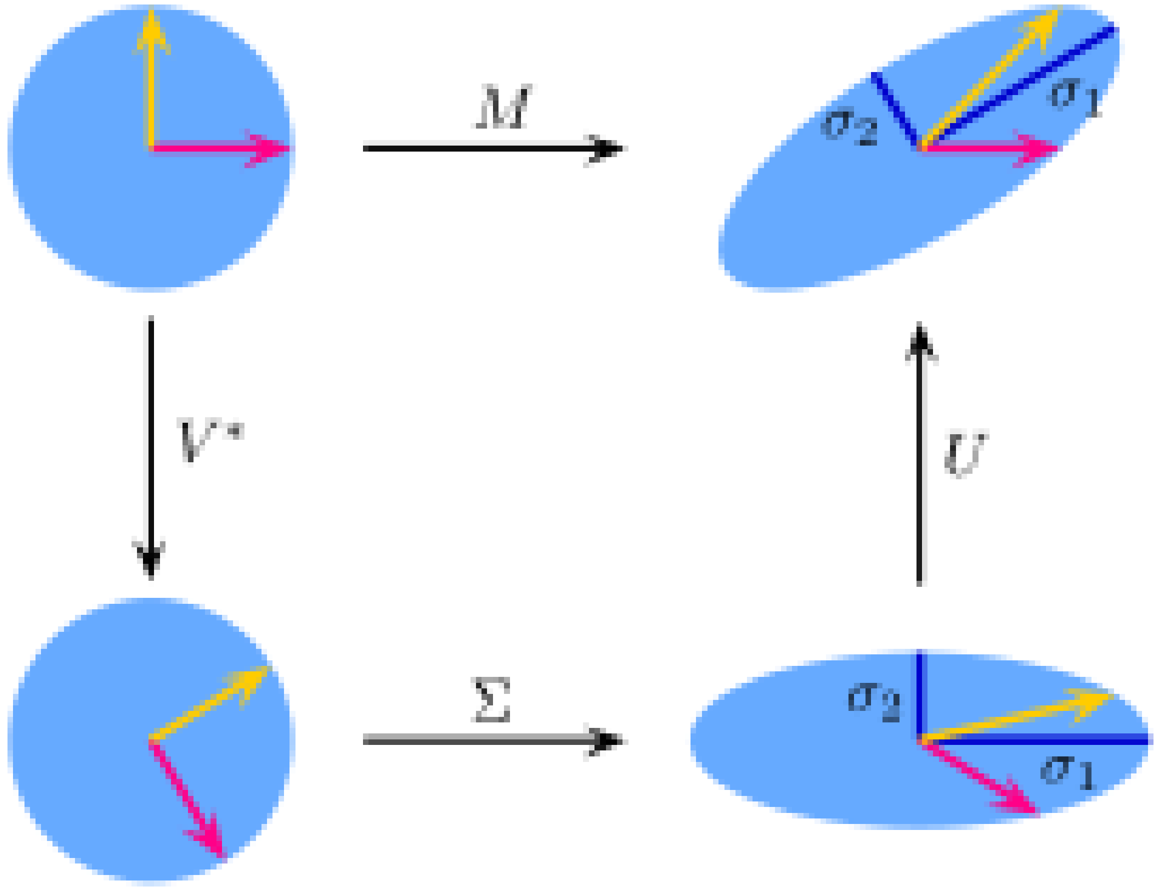
Декомпозиция матриц SVD

Сингулярное разложение (Singular value decomposition)

$$A = U\Sigma V^T$$

$$A_k = U_k \Sigma_k V_k^T = (U_k \Sigma_k) V_k^T = U_k (\Sigma_k V_k^T)$$





$$M = U \cdot \Sigma \cdot V^*$$

$$U \in \mathbb{R}^{m \times r}; UU^T = I$$

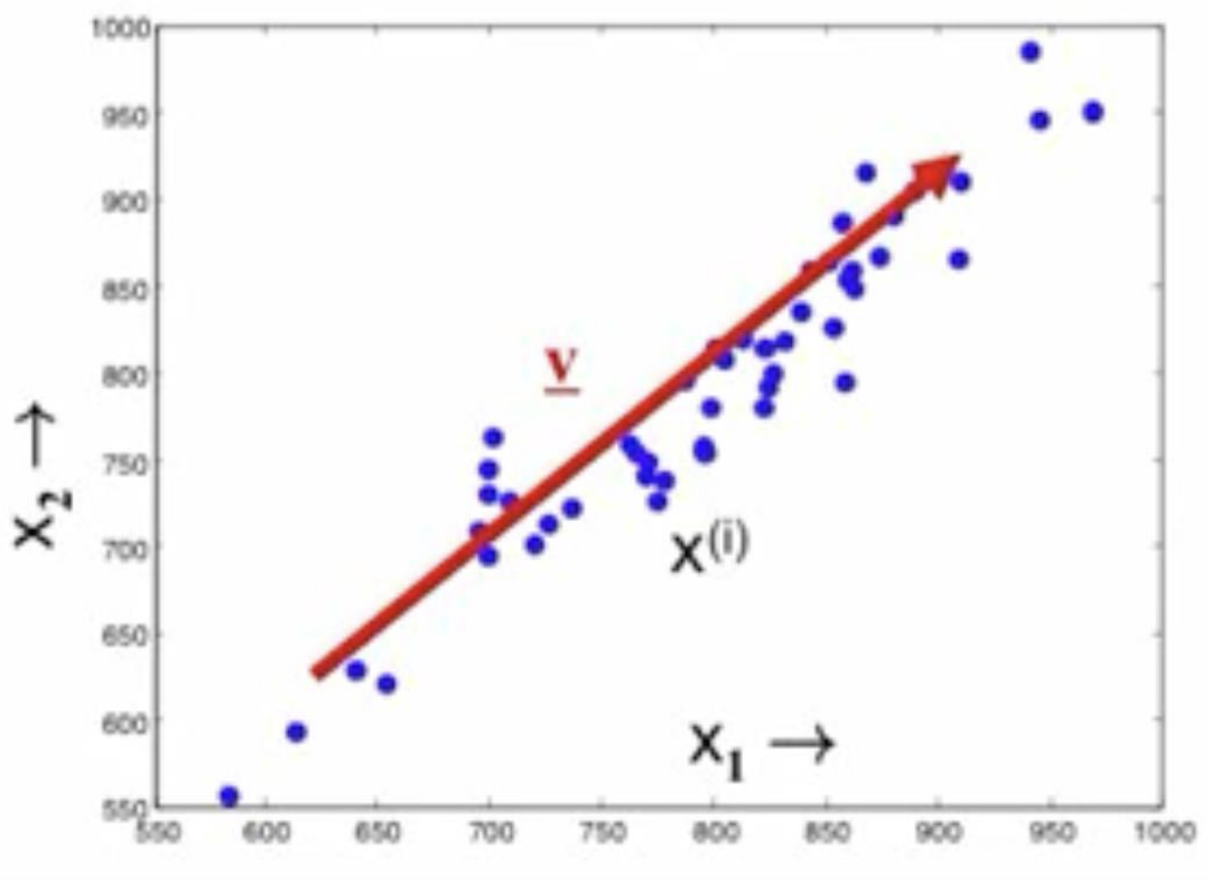
$$V \in \mathbb{R}^{n \times r}; VV^T = I$$

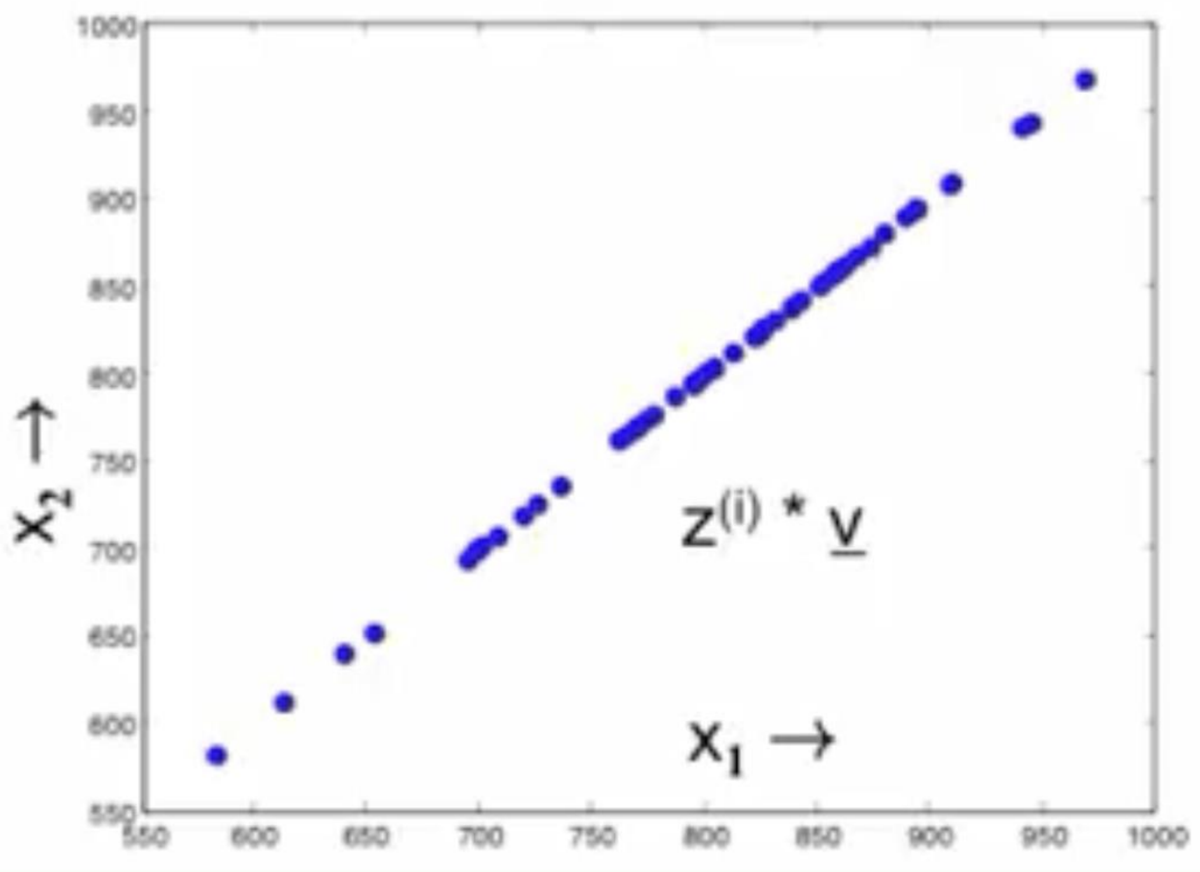
$$\Sigma = diag(\sigma_1, ..., \sigma_r); r = rank(M)$$

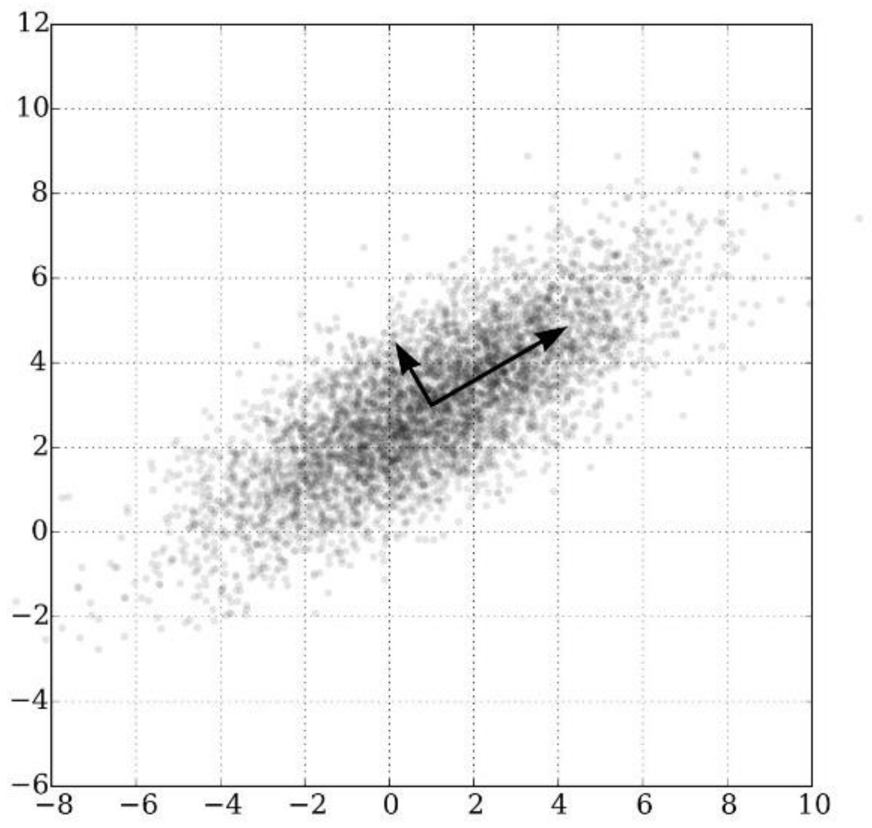
Principal component analysis PCA

Построим новый базис f(z): [x1, x2, …, xn] = z * v = z * [v1, v2, …, vn]
где v будет направлением максимальной дисперсии по x1, x2

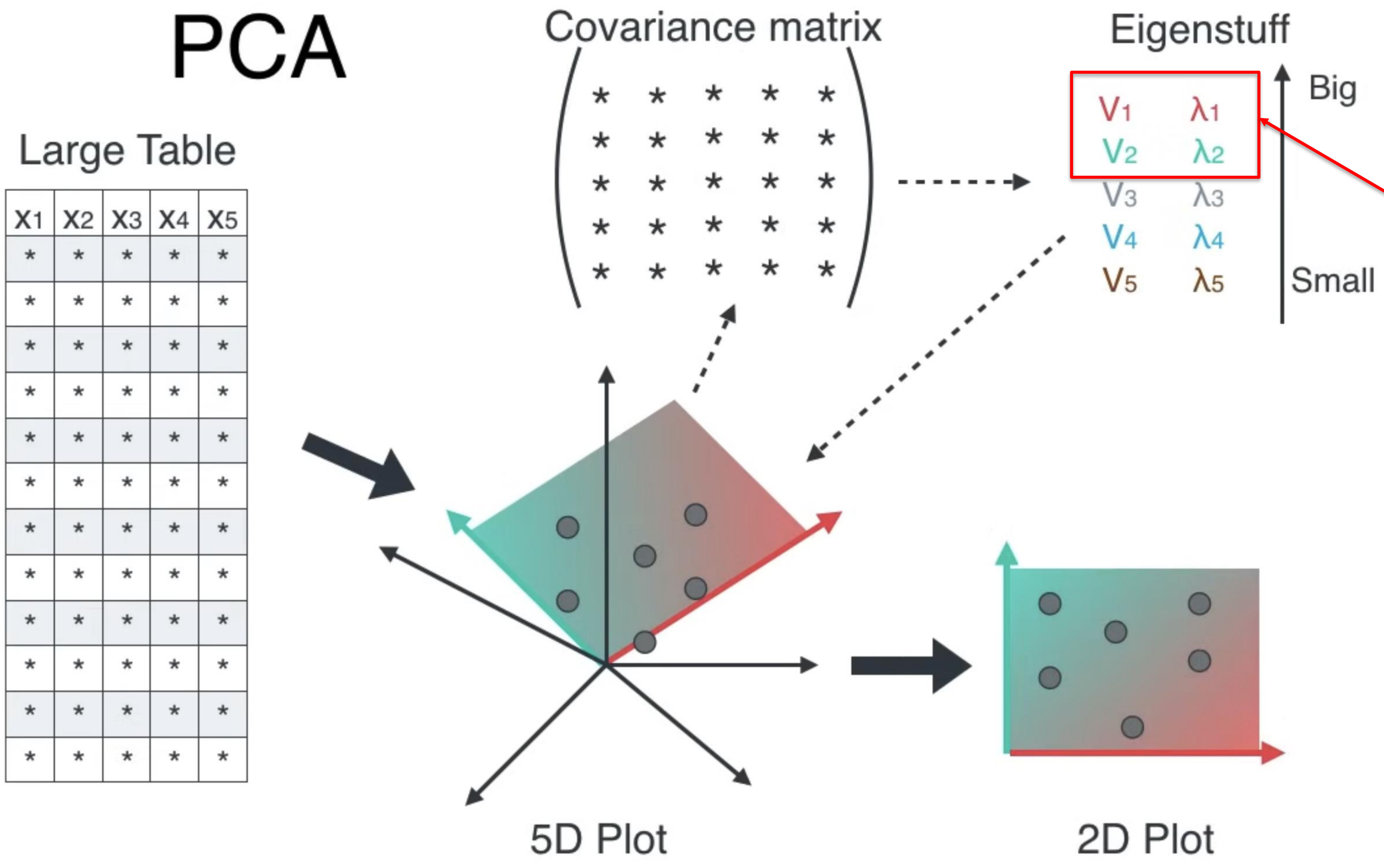
Выберем новые оси дисперсия по осям была максимальной. Это эквивалентно поиску ортогонального базиса в пространстве дисперсий, то есть решению $XX^Te = \lambda e$
где e – ортогональный базис и λ – собственные значения матрицы ковариации XX^T







Алгоритм PCA



Отсортированы по значению собственного числа

Top@2 – самых значимых компонент

SVD и PCA

Factorization of Data Matrix F:

$$F_{N\times M} = U_{N\times N} \cdot \Sigma_{N\times M} \cdot {V^T}_{M\times M}$$

Covariance Matrix: $R = FF^T$

$$R = U \cdot \Sigma \cdot \underbrace{V^T \cdot V}_{I} \cdot \Sigma^T \cdot U^T$$

$$R = U \cdot \Lambda \cdot U^T$$

Factorization of Data Matrix F:

$$R_{N\times N} = U_{N\times N} \cdot \Lambda_{N\times N} \cdot {U^T}_{N\times N}$$

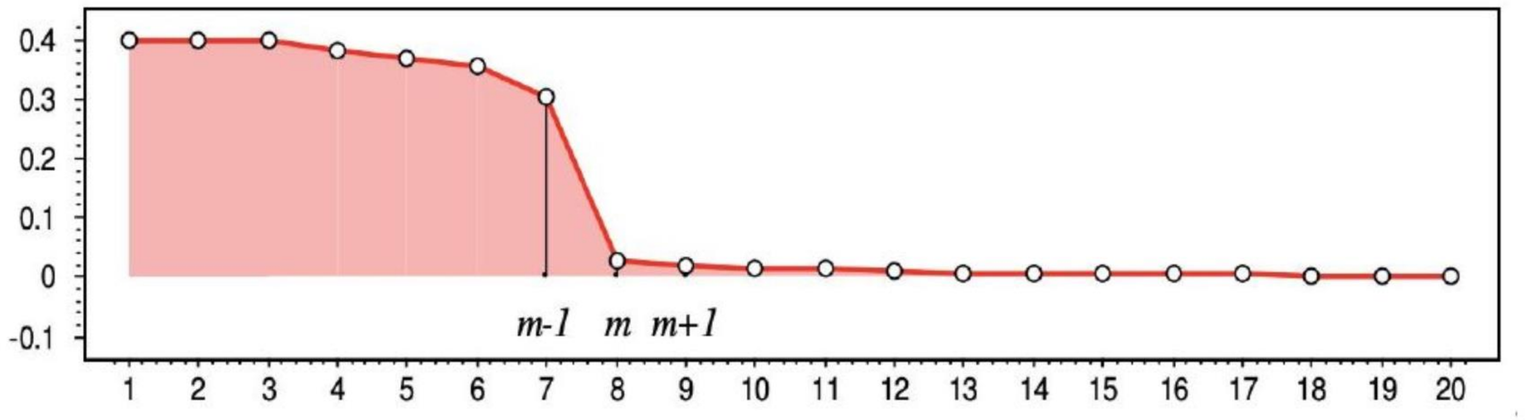
Eigenvalues:

$$\Lambda_{N\times N} = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_4 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & \lambda_K & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & \lambda_{K+1} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \lambda_N \end{pmatrix} \quad \lambda_i = {\sigma_i}^2$$

https://education.yandex.ru/handbook/data-analysis/article/metod-glavnyh-komponent-i-faktornyj-analiz

Понижение размерности PCA

- Часто данные содержат помехи и неинформативны
- Удалим компоненты с низкой дисперсией в PCA



$$E_m = \frac{\|GU^{\mathsf{T}} - F\|^2}{\|F\|^2} = \frac{\lambda_{m+1} + \cdots + \lambda_n}{\lambda_1 + \cdots + \lambda_n} \leqslant \varepsilon.$$

Top16

Top50

Место для ваших вопросов