МАШИННОЕ ОБУЧЕНИЕ И АНАЛИЗ ДАННЫХ

(Machine Learning and Data Mining)

Н. Ю. Золотых

http://www.uic.unn.ru/~zny/ml

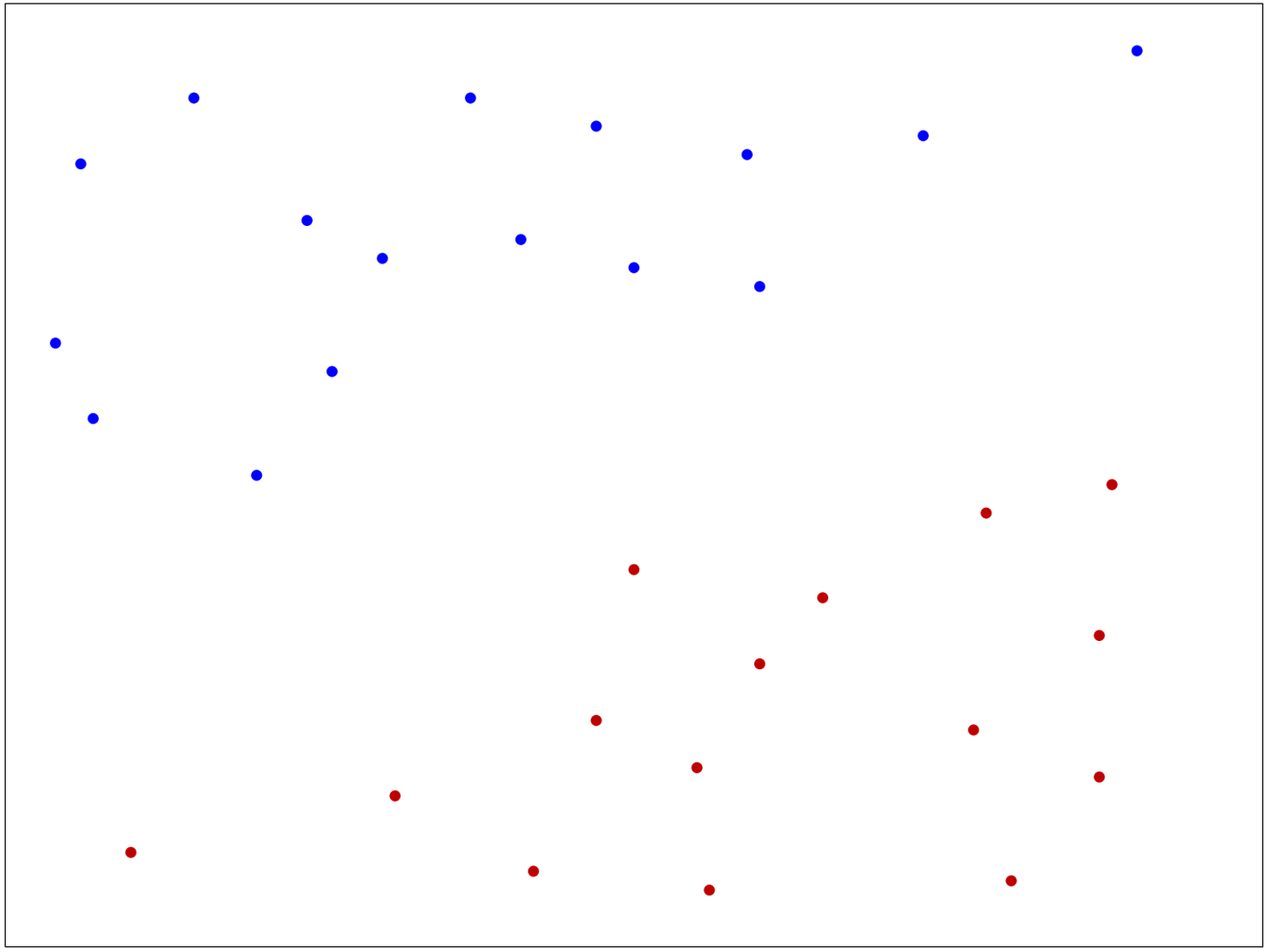
Лекция 12

Машина опорных векторов

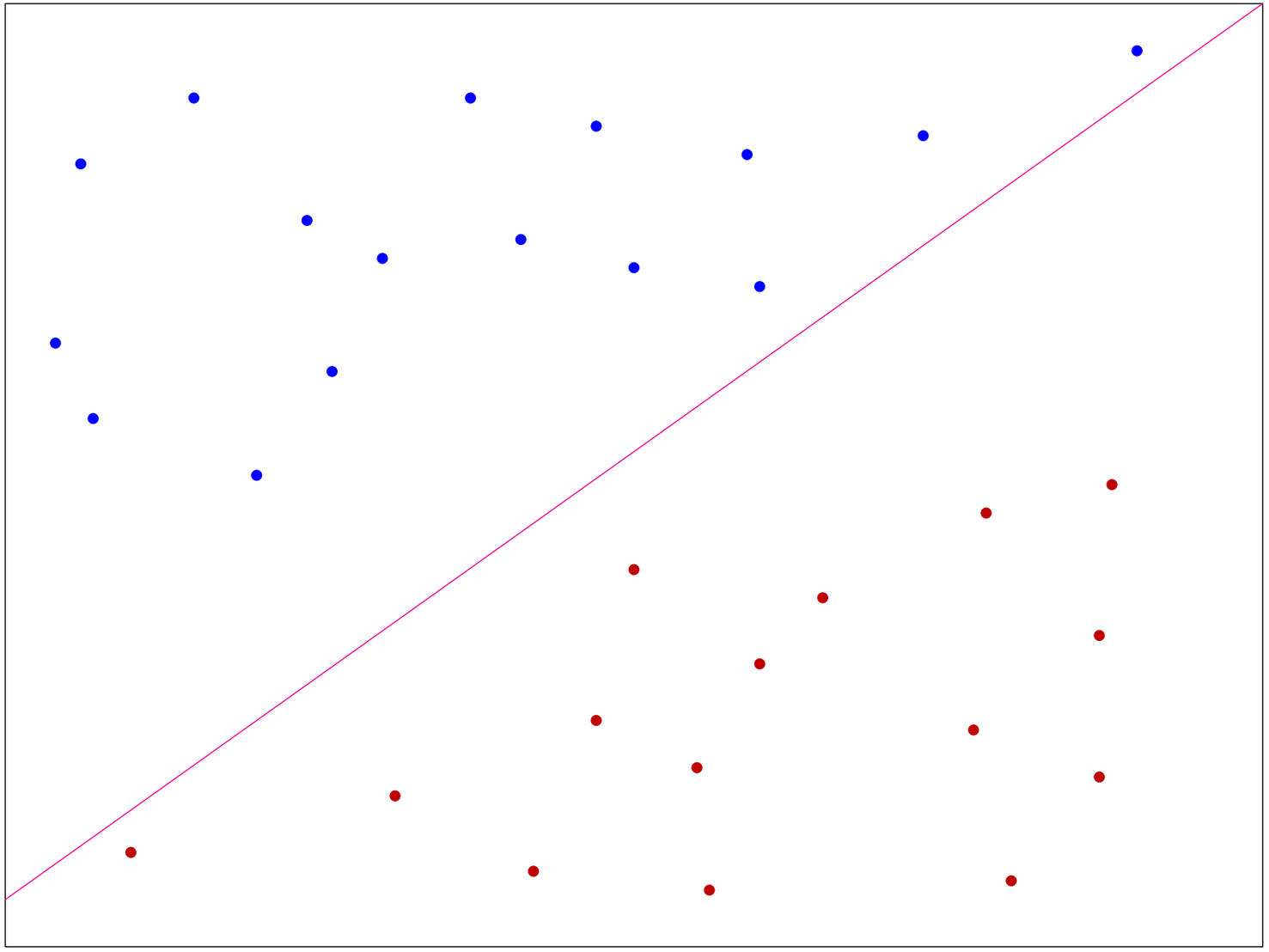
SVM — Support Vector Machine

- Метод обобщенного портрета (оптимальная разделяющая гиперплоскость) — 60–70 гг. В. Н. Вапник и др., см. В. Н. Вапник, А. Я. Червоненкис «Теория распознавания образов». М.: Наука, 1974
- Добавлены ядра — [Cortes, Vapnik, 1995]

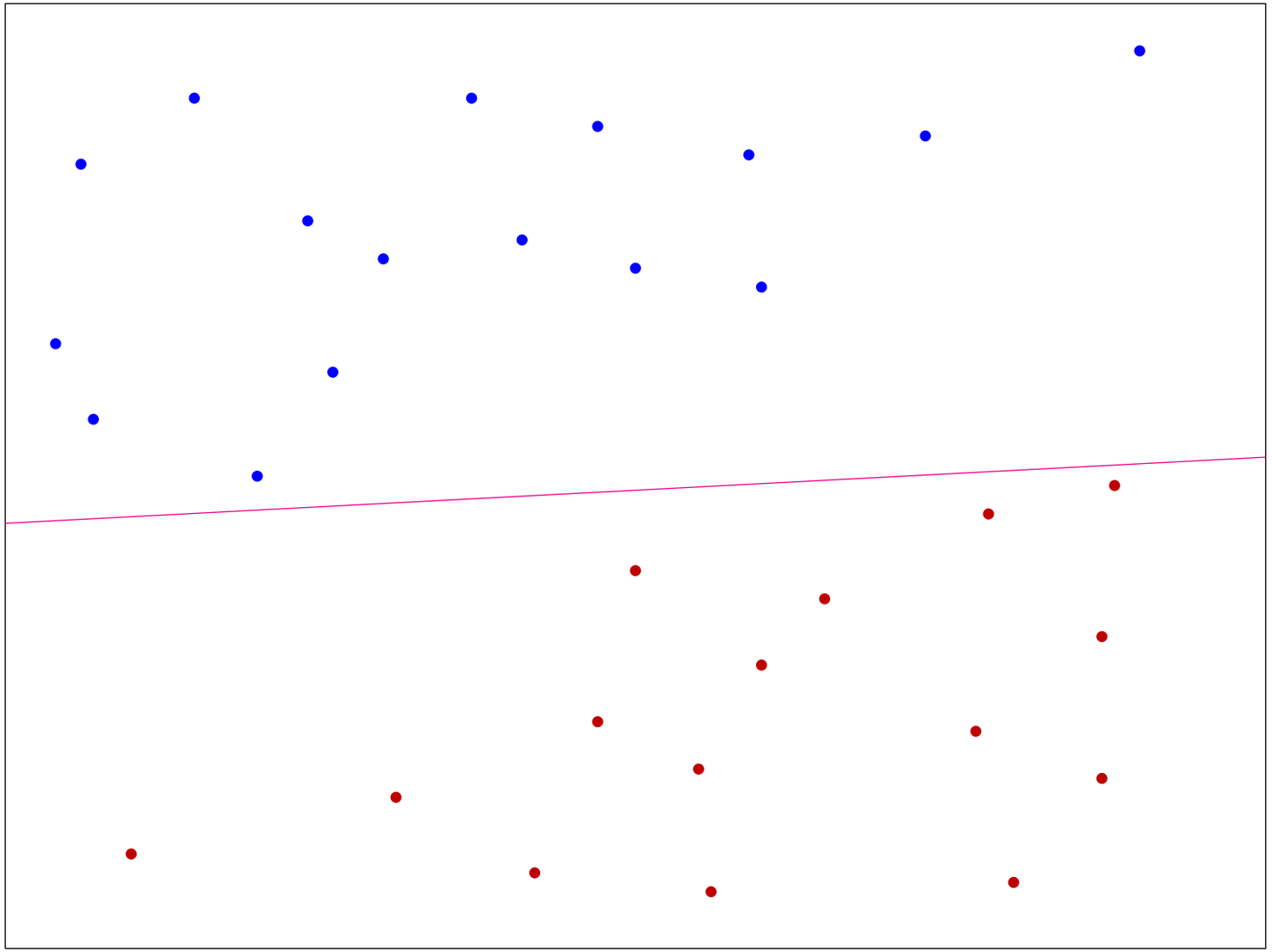
12.1. Оптимальная разделяющая гиперплоскость



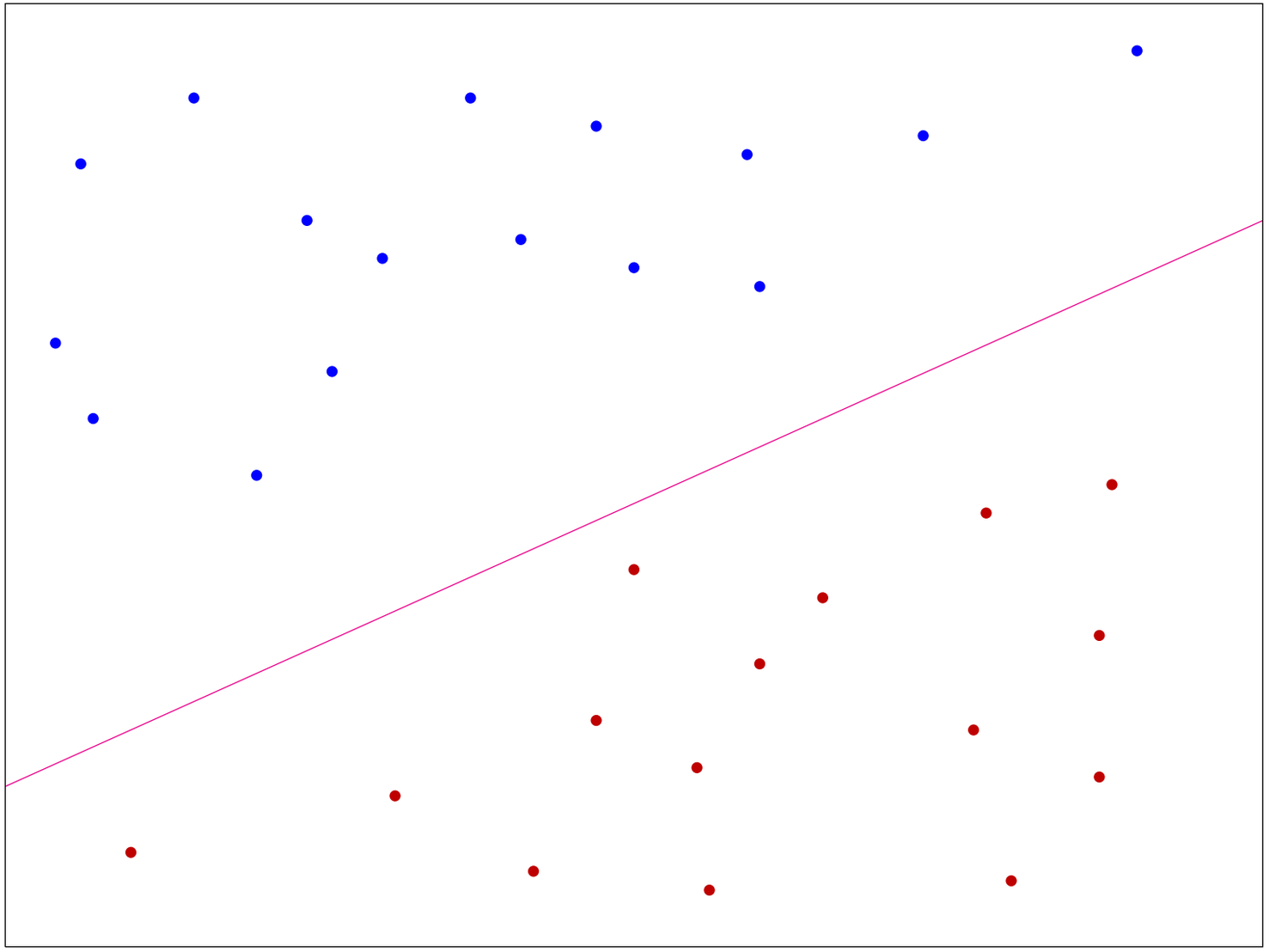
Два класса



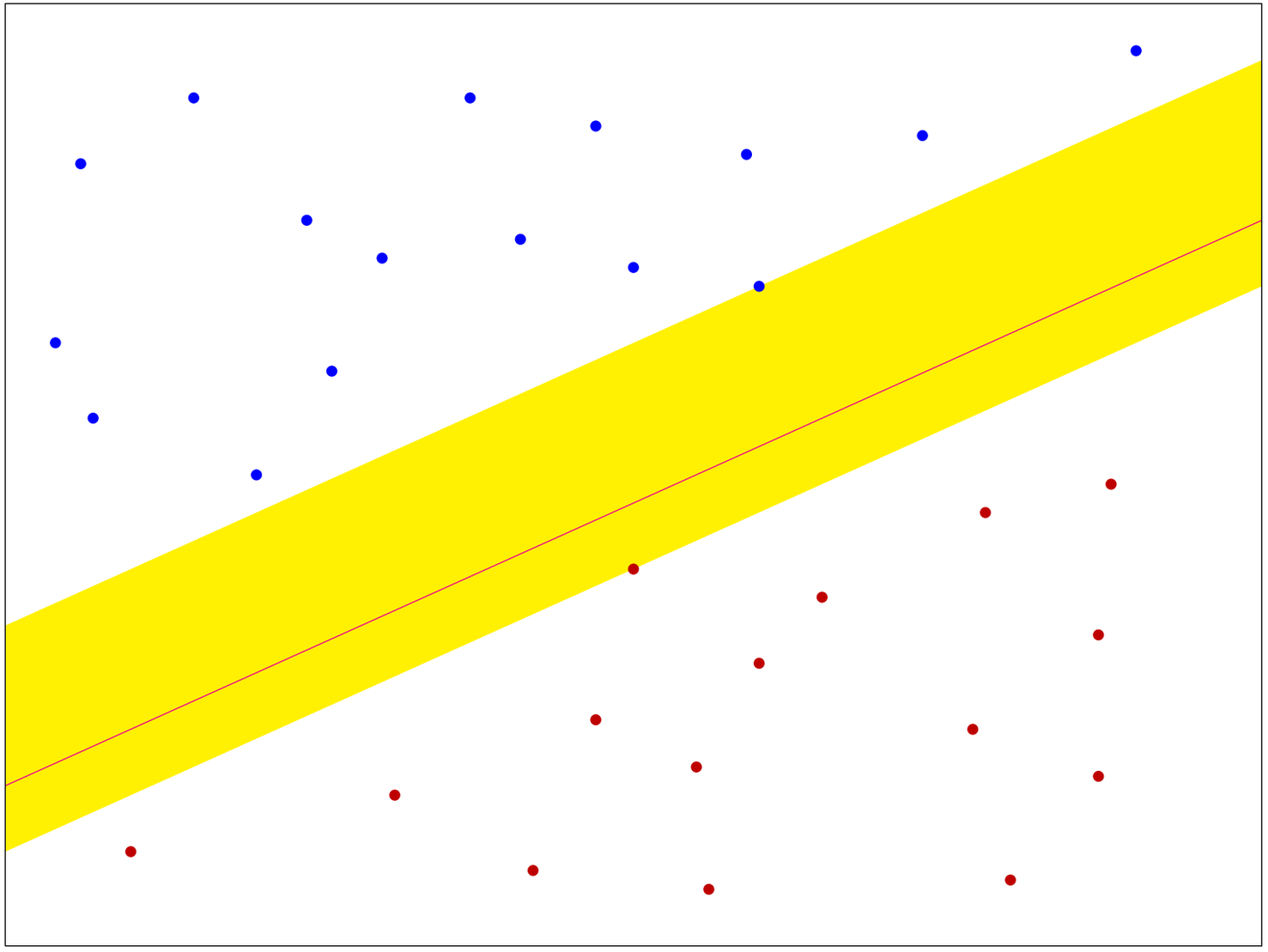
Одна из возможных разделяющих прямых



Другая из возможных разделяющих прямых



Еще одна из возможных разделяющих прямых (гиперплоскостей)

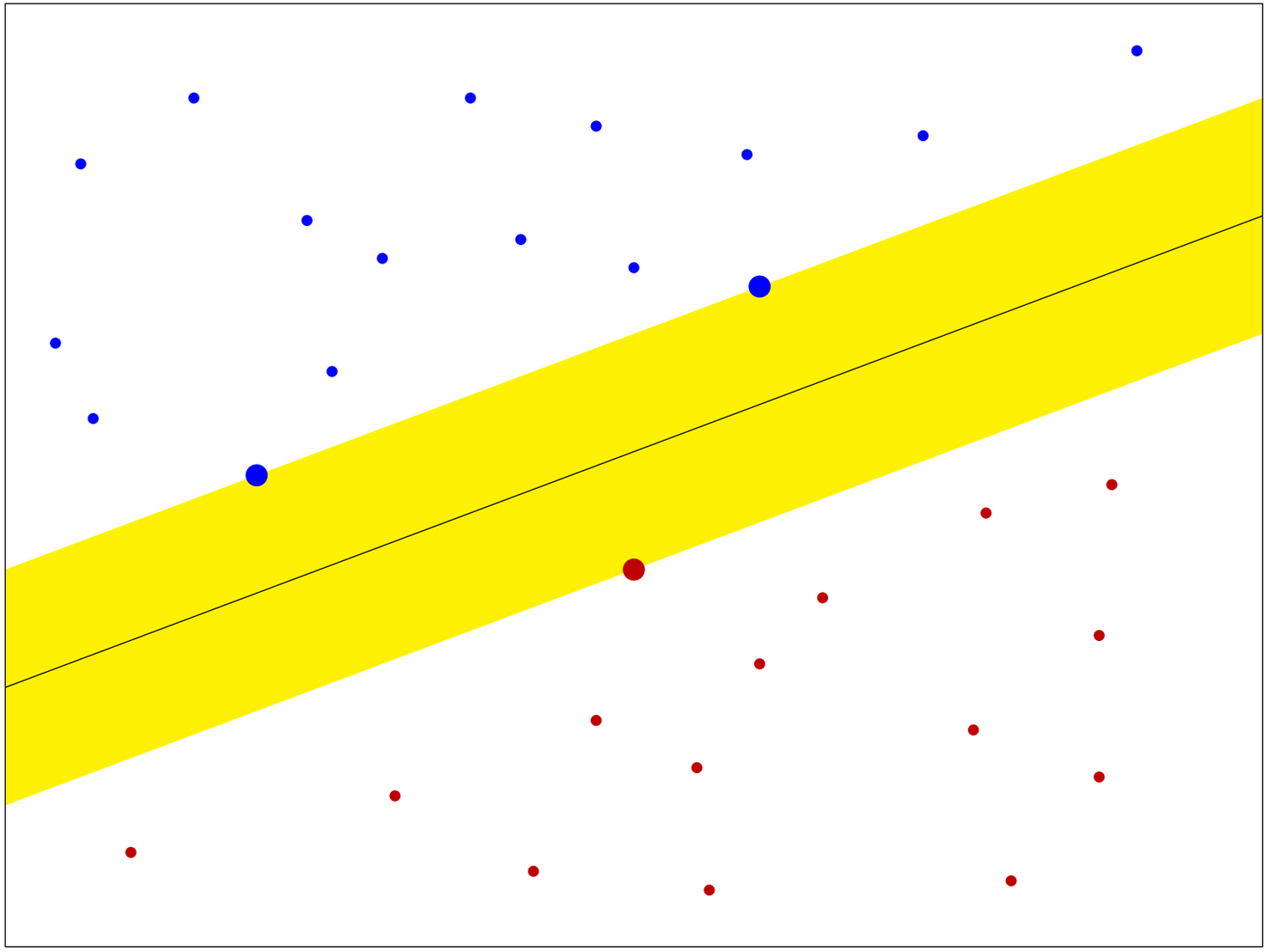


Желтая — разделяющая («нейтральная») полоса

Оптимальная разделяющая гиперплоскость — это гиперплоскость, максимизирующая ширину разделяющей полосы и лежащая в середине этой полосы.

Иными словами, оптимальная разделяющая гиперплоскость максимизирует *зазор* (*margin*) между плоскостью и данными из обучающей выборки.

Если классы линейно разделимы и каждый содержит не менее одного элемента, то оптимальная разделяющая гиперплоскость единственна.



Задача поиска оптимальной гиперплоскости эквивалентна следующей:

$$\max_{\beta,\,\beta_0} C$$

при ограничениях

$$y^{(i)}(\beta^\top x^{(i)} + \beta_0) \geq C \quad (i = 1, 2, \ldots, N)$$

$$\|\beta\| = 1$$

или, что эквивалентно,

$$\min_{\beta,\,\beta_0} \frac{1}{2}\|\beta\|^2$$

при ограничениях

$$y^{(i)}(\beta^\top x^{(i)} + \beta_0) \geq 1 \quad (i = 1, 2, \ldots, N).$$

Во втором случае зазор равен $2/\|\beta\|$.

Получили задачу выпуклого программирования (минимизация квадратичной функции при линейных ограничениях).

Функция Лагранжа:

$$\mathscr{L}(\beta_0, \beta, \lambda) = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \lambda_i \Big(y^{(i)}(\beta^\top x^{(i)} + \beta_0) - 1 \Big).$$

Так как задача выпуклая, то

$$\underbrace{\min_{\beta,\,\beta_0} \overbrace{\max_{\lambda \geq 0} \mathscr{L}(\beta_0, \beta, \lambda)}^{v(\beta, \beta_0)}}_{\text{прямая задача}} = \underbrace{\max_{\lambda \geq 0} \overbrace{\min_{\beta,\,\beta_0} \mathscr{L}(\beta_0, \beta, \lambda)}^{w(\lambda)}}_{\text{двойственная задача}}$$

Решаем двойственную задачу.

$$\frac{\partial \mathscr{L}(\beta_0, \beta, \lambda)}{\partial \beta_0} \quad \Rightarrow \quad 0 = \sum_{i=1}^{N} \lambda_i y^{(i)}, \qquad \frac{\partial \mathscr{L}(\beta_0, \beta, \lambda)}{\partial \beta} \quad \Rightarrow \quad \beta = \sum_{i=1}^{N} \lambda_i y^{(i)} x^{(i)}.$$

Двойственная функция:

$$w(\lambda) = \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \lambda_i \lambda_k y^{(i)} y^{(k)} {x^{(i)}}^\top x^{(k)} - \sum_{i=1}^{N} \lambda_i \;\to\; \min_{\lambda \geq 0}.$$

На практике обычно решают именно эту — двойственную — задачу.

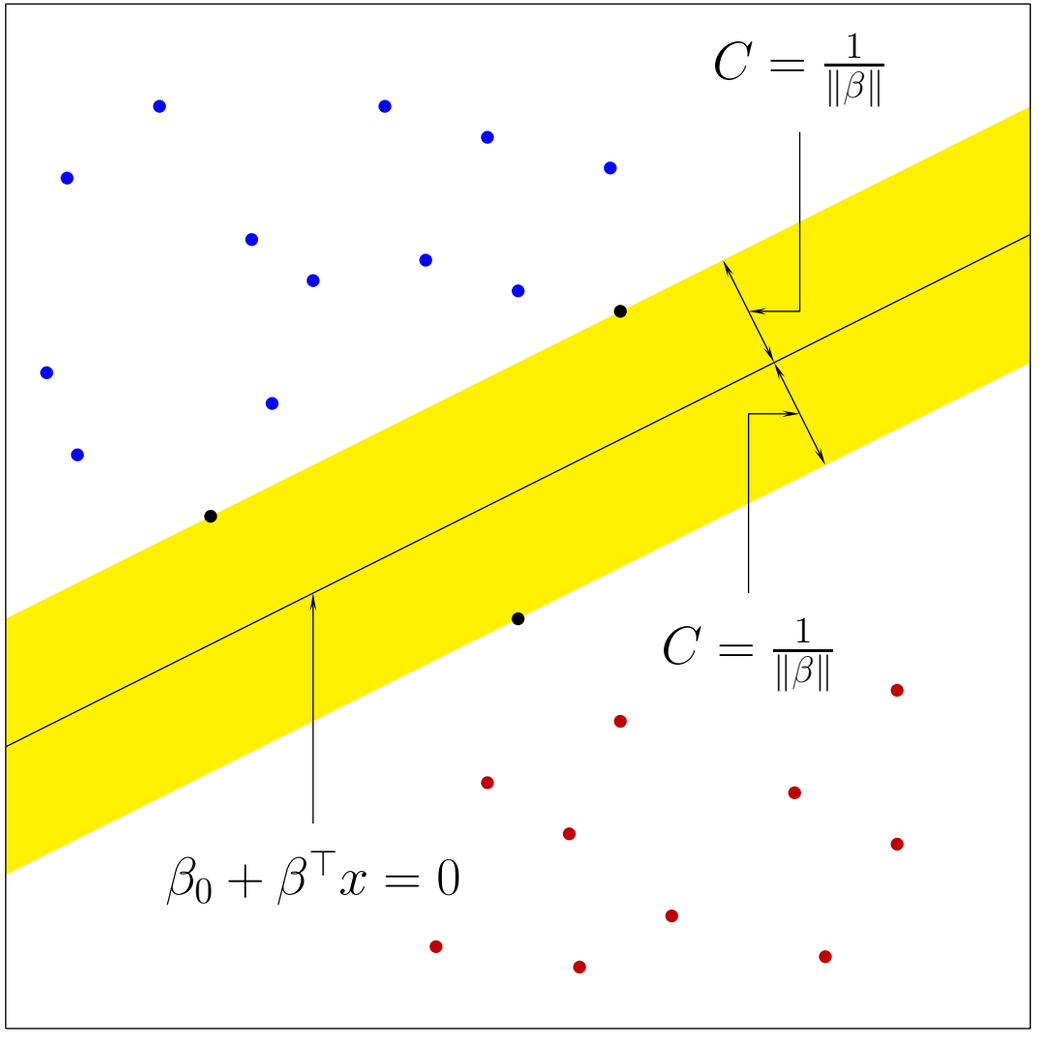
Решение задачи удовлетворяет условию Куна–Таккера (*дополняющей нежесткости*):

$$\lambda_i \Big(y^{(i)}(\beta^\top x^{(i)} + \beta_0) - 1 \Big) = 0 \qquad (i = 1, 2, \ldots, N),$$

откуда следует, что

- если $\lambda_i > 0$, то $y^{(i)}(\beta^\top x^{(i)} + \beta_0) = 1$, т. е. $x^{(i)}$ лежит на границе разделяющей полосы
- если $y^{(i)}(\beta^\top x^{(i)} + \beta_0) > 1$, т. е. $x^{(i)}$ не лежит на границе разд. полосы, то $\lambda_i = 0$

Точки, для которых $y^{(i)}(\beta^\top x^{(i)} + \beta_0) = 1$, называются *опорными точками* или *опорными векторами*.


$C = \frac{1}{\|\beta\|}$
$C = \frac{1}{\|\beta\|}$
$\beta_0 + \beta^\top x = 0$

По построению, ни один объект из обучающей выборки не может попасть внутрь разделяющей полосы. Однако в нее могут попасть тестовые объекты. Тем не менее, даже в этом случае решающее правило, основанное на оптимальной разделяющей гиперплоскости, показывает, как правило, хорошие результаты.

Расположение оптимальной разделяющей гиперплоскости полностью определяется только опорными точками и не зависит от других точек обучающей выборки. Такая робастность отличает метод, основанный на оптимальных разделяющих гиперплоскостях, от *LDA*, в котором учитываются даже точки, расположенные далеко от границы. Конечно, для нахождения самих опорных точек мы должны принимать во внимание все точки обучающей выборки. Однако если данные действительно подчиняются нормальному закону с равными матрицами ковариации, то *LDA* — лучший выбор.

Разделяющая гиперплоскость, найденная с помощью логистической регрессии, часто близка к оптимальной разделяющей гиперплоскости. Это можно объяснить некоторой схожестью этих походов: на логистическую регрессии можно смотреть как на взвешенный метод наименьших квадратов, причем веса тем больше, чем ближе точка к границе.

Что делать, если данные линейно не разделимы?

12.2. Случай «перекрывающихся» классов

Предположим, что классы линейно-неразделимы. Рассмотрим задачу

$$\max_{\beta,\,\beta_0,\,\xi_i} C$$

при ограничениях

$$\|\beta\| = 1, \quad y^{(i)}(\beta^\top x^{(i)} + \beta_0) \geq C(1 - \xi_i), \quad \xi_i \geq 0 \quad (i = 1, 2, \ldots, N), \quad \sum_{i=1}^{n} \xi_i \leq \Xi,$$

где Ξ — некоторая константа (параметр метода).

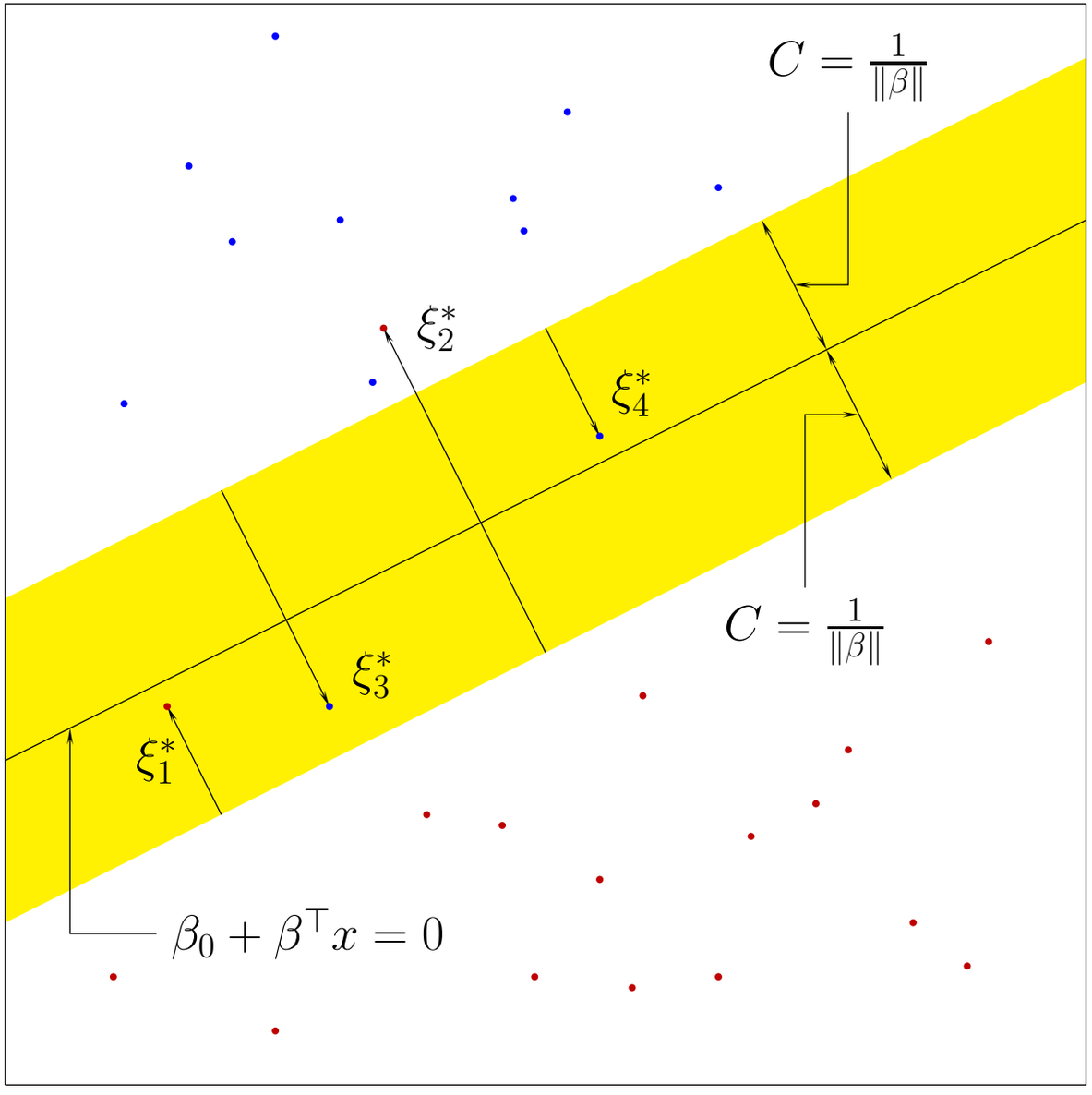
Случаю линейно-отделимых классов соответствует $\Xi = 0$.

ξ_i пропорционально расстоянию, на которое $x^{(i)}$ заходит за границу разделяющей полосы.

В частности, i-й объект будет классифицирован не правильно $\Leftrightarrow \xi_i > 1$.

Чем меньше Ξ, тем меньше объектов будет классифицировано неправильно.

Но параметр Ξ должен быть достаточно велик, чтобы задача была совместной.


$C = \frac{1}{\|\beta\|}$
ξ_2^*
ξ_4^*
ξ_3^*
ξ_1^*
$C = \frac{1}{\|\beta\|}$
$\beta_0 + \beta^\top x = 0$

Эквивалентная запись задачи:

$$\min_{\beta,\,\beta_0,\,\xi_i} \frac{1}{2}\|\beta\|^2,$$

при ограничениях

$$y^{(i)}(\beta^\top x^{(i)} + \beta_0) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (i = 1, 2, \ldots, N), \quad \sum_{i=1}^{n} \xi_i \leq \Xi.$$

Задача заключается в минимизации квадратичной положительно определенной функции при линейных ограничениях.

Запишем задачу в виде

$$\min_{\beta,\,\beta_0,\,\xi_i} \frac{1}{2}\|\beta\|^2 + \gamma \sum_{i=1}^{n} \xi_i,$$

при ограничениях

$$y_i(\beta^\top x^{(i)} + \beta_0) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (i = 1, 2, \ldots, N),$$

где $\gamma = 1/\Xi$. Случай линейно отделимых областей соответствует значению $\gamma = \infty$.

Функция Лагранжа для этой задачи имеет вид

$$\mathscr{L}(\beta_0, \beta, \xi, \lambda, \mu) = \frac{1}{2}\|\beta\|^2 + \gamma \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{N} \lambda_i \Big(y^{(i)}(\beta^\top x^{(i)} + \beta_0) - (1 - \xi_i) \Big) - \sum_{i=1}^{N} \mu_i \xi_i.$$

Положим производные относительно неизвестных β_0, β, ξ_i равными нулю:

$$0 = \sum_{i=1}^{N} \lambda_i y^{(i)}, \quad \beta = \sum_{i=1}^{N} \lambda_i y^{(i)} x^{(i)}, \quad \lambda_i = \gamma - \mu_i \qquad (i = 1, 2, \ldots, N).$$

Подставляя эти формулы в $\mathscr{L}(\beta_0, \beta, \xi, \lambda, \mu)$, получим двойственную функцию Лагранжа

$$w(\lambda) = \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \lambda_i \lambda_k y^{(i)} y^{(k)} {x^{(i)}}^\top x^{(k)} - \sum_{i=1}^{N} \lambda_i.$$

$w(\lambda)$ необходимо минимизировать при ограничениях

$$0 \le \lambda_i \le \gamma, \qquad \sum_{i=1}^{N} \lambda_i y^{(i)} = 0.$$

Условия Куна–Таккера (дополняющей нежесткости):

$$\lambda_i\Big(y^{(i)}(\beta^\top x^{(i)} + \beta_0) - (1 - \xi_i)\Big) = 0,$$

$$\mu_i \xi_i = 0$$

Пусть β, β_0, ξ_i, λ_i и т. д. — оптимальные значения соответствующих неизвестных.

Если $\lambda_i \neq 0$, то $y^{(i)}(\beta^\top x^{(i)} + \beta_0) = 1 - \xi_i$, т. е. i-е неравенство выполнено как равенство.

Таким образом, в формуле

$$\beta = \sum_{i=1}^{N} \lambda_i y^{(i)} x^{(i)},$$

в правой части остаются только слагаемые, соответствующие точкам, для которых $y^{(i)}({x^{(i)}}^\top \beta + \beta_0) = 1 - \xi_i$.

Эти точки (объекты) называются *опорными векторами*, так как β зависит только от них.

Среди этих точек некоторые могут лежать на границе разделяющей полосы ($\xi_i = 0$).

Для них $0 < \lambda_i < \gamma$. Для остальных опорных точек $\xi_i > 0$ и $\lambda_i = \gamma$.

Любая точка на границе разделяющей полосы может использоваться для определения β_0. На практике в качестве β_0 берется среднее из всех значений, определяемых по опорным векторам.

Итак, на расположение разделяющей гиперплоскости влияют только опорные точки.

Это выгодно отличает данный классификатор от *LDA*, в котором граница областей определяется матрицами ковариации и расположением центроидов, и, следовательно, зависит от всех точек.

В рассматриваемом отношении классификатор опорных векторов больше похож на логистическую регрессию.

Train error = 0.195
Test error = 0.235
Bayes error = 0.21
x_2
1.5
1.0
0.5
0.0

12.3. Ядра и спрямляющие пространства

Перейдем от исходного пространства $\mathscr{X}$ в другое, называемое *спрямляющим*, $\mathscr{H}$ с помощью некоторого отображения

$$h(x) = \Big(h_1(x), \ldots, h_M(x)\Big),$$

где $h_m(x)$ — базисные функции (новые признаки) $(m = 1, 2, \ldots, M)$.

Новый классификатор определяется теперь функцией

$$f(x) = \operatorname{sign}\left(\beta^\top h(x) + \beta_0\right).$$

В формуле

$$\beta = \sum_{i=1}^{N} \lambda_i y^{(i)} x^{(i)},$$

заменяем $x^{(i)}$ на $h(x^{(i)})$:

$$\beta = \sum_{i=1}^{N} \lambda_i y^{(i)} h(x^{(i)}),$$

Двойственная функция Лагранжа

$$L_D = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j y^{(i)} y^{(j)} {x^{(i)}}^{\top} x^{(j)} - \sum_{i=1}^{N} \lambda_i.$$

примет вид

$$L_D = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j y^{(i)} y^{(j)} \left\langle h(x^{(i)}), h(x^{(j)}) \right\rangle - \sum_{i=1}^{N} \lambda_i.$$

Функция $f(x)$ запишется как

$$f(x) = \text{sign}\left(\beta^{\top} h(x) + \beta_0\right) = \text{sign}\left(\sum_{i=1}^{N} \lambda_i y^{(i)} \left\langle h(x), h(x^{(i)}) \right\rangle + \beta_0\right).$$

Мы видим, что $h(x)$ встречается только в скалярном произведении $\left\langle h(x), h(x^{(i)}) \right\rangle$!

Таким образом, для определения классификатора опорных векторов нам достаточно уметь вычислять лишь функцию

$$K(x, x') = \langle h(x), h(x') \rangle .$$

Итак, мы можем заменить скалярное произведение функцией $K(x, x')$ и, более того, вообще явно не строить спрямляющего пространства $\mathscr{H}$, а подбирать функцию K.

Можно совсем отказаться от построения новых признаков, а попробовать построить модель, в которой описываются взаимоотношения между объектами с помощью функции $K(x, x')$.

Рассмотрим необходимые и достаточные условия, которым должна удовлетворять функция $K(x, x')$.

Функция $K : \mathscr{X} \times \mathscr{X} \to \mathbf{R}$ называется *ядром*, если она представима в виде $K(x, x') = \langle h(x), h(x') \rangle$ при некотором отображении $h : \mathscr{X} \to \mathscr{H}$, где $\mathscr{H}$ — евклидово (или гильбертово) пространство со скалярным произведением $\langle \cdot, \cdot \rangle$.

Теорема 12.1 (Мерсер) *Функция $K(x, x')$ является ядром тогда и только тогда, когда она симметрична, т. е. $K(x, x') = K(x', x)$, и неотрицательно определена, т. е.*

$$\int_{\mathscr{X}} \int_{\mathscr{X}} K(x, x') g(x) g(x') dx dx' \geq 0$$

для всех $g(x)$, для которых $\int_{\mathscr{X}} g(x)^2 dx$ ограничено.

Пример

Рассмотрим пространство признаков размерности 2 с двумя входами x_1, x_2 и полиномиальным ядром степени 2:

$$K(x, x') = (1 + \langle x, x' \rangle)^2 = (1 + x_1 x'_1 + x_2 x'_2)^2 = 1 + 2x_1 x'_1 + 2x_2 x'_2 + (x_1 x'_1)^2 + (x_2 x'_2)^2 + 2x_1 x'_1 x_2 x'_2.$$

Мы видим, что $M = 6$ и можно положить

$$h_1(x) = 1,\ h_2(x) = \sqrt{2}x_1,\ h_3(x) = \sqrt{2}x_2,\ h_4(x) = x_1^2,\ h_5(x) = x_2^2,\ h_6(x) = \sqrt{2}x_1 x_2.$$

Тогда $K(x, x') = \langle h(x), h(x') \rangle$.

Примеры ядер

- $K(x, x') = 1$
- $K(x, x') = \langle x, x' \rangle$
- произведение ядер $K(x, x') = K_1(x, x')K_2(x, x')$ — ядро
- $K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x')$ для любых $\alpha_1 > 0$, $\alpha_2 > 0$
- $K(x, x') = \varphi(x)\varphi(x')$ для любой $\varphi : \mathbf{R}^d \to \mathbf{R}$
- $K(x, x') = K_0(\varphi(x), \varphi(x'))$ для любой $\varphi : \mathbf{R}^d \to \mathbf{R}^d$
- $K(x, x') = \int_{\mathscr{X}} s(x, z)s(x', z)dz$, где $s : \mathscr{X} \times \mathscr{X} \to \mathbf{R}$ – симметричная интегрируемая функция
- если K_0 – ядро и $\varphi : \mathbf{R} \to \mathbf{R}$ представима в виде сходящегося степенного ряда с неотрицательными коэффициентами, то $K(x, x') = \varphi(K_0(x, x'))$ – ядро

Примеры ядер

- однородный многочлен степени m: $K(x, x') = \langle x, x' \rangle^m$,
- неоднородный многочлен степени m: $K(x, x') = (1 + \langle x, x' \rangle)^m$,
- радиальная функция: $K(x, x') = e^{-\gamma \|x - x'\|^2}$ (предел многочленов при $m \to \infty$; H– бесконечномерное),
- сигмоидальная (масштабированная логистическая) функция:

$$K(x, x') = \tanh(\kappa_1 \langle x, x' \rangle + \kappa_2), \qquad \text{где} \quad \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

SVM – аналог kNN

$$f(x) = \text{sign}\left(\beta^\top h(x) + \beta_0\right) = \text{sign}\left(\sum_{i=1}^{N} \lambda_i y^{(i)} \left\langle h(x), h(x^{(i)}) \right\rangle + \beta_0\right).$$

(напомним, что $\lambda_i \neq 0 \Leftrightarrow x^{(i)}$ – опорный)

Исследуем роль параметра γ в

$$\min_{\beta,\,\beta_0,\,\xi_i} \frac{1}{2}\|\beta\|^2 + \gamma \sum_{i=1}^{n} \xi_i,$$

учитывая, что классы в спрямляющем пространстве, как правило, можно разделить.

Большое значение γ препятствует росту ξ_i, что, как правило, приводит к переобучению и крайне извилистой границе между областями.

Маленькое значение γ способствует росту ξ_i и обычно приводит к более «прямой» разделяющей поверхности.

На SVM можно смотреть как на регуляризованный метод минимизации эмпирического риска, если в качестве штрафной функции рассматривается

$$L(g,\, y) = \Big[1 - yg\Big]_+,$$

где

$f(x) = \operatorname{sign} g(x)$ — классификатор,

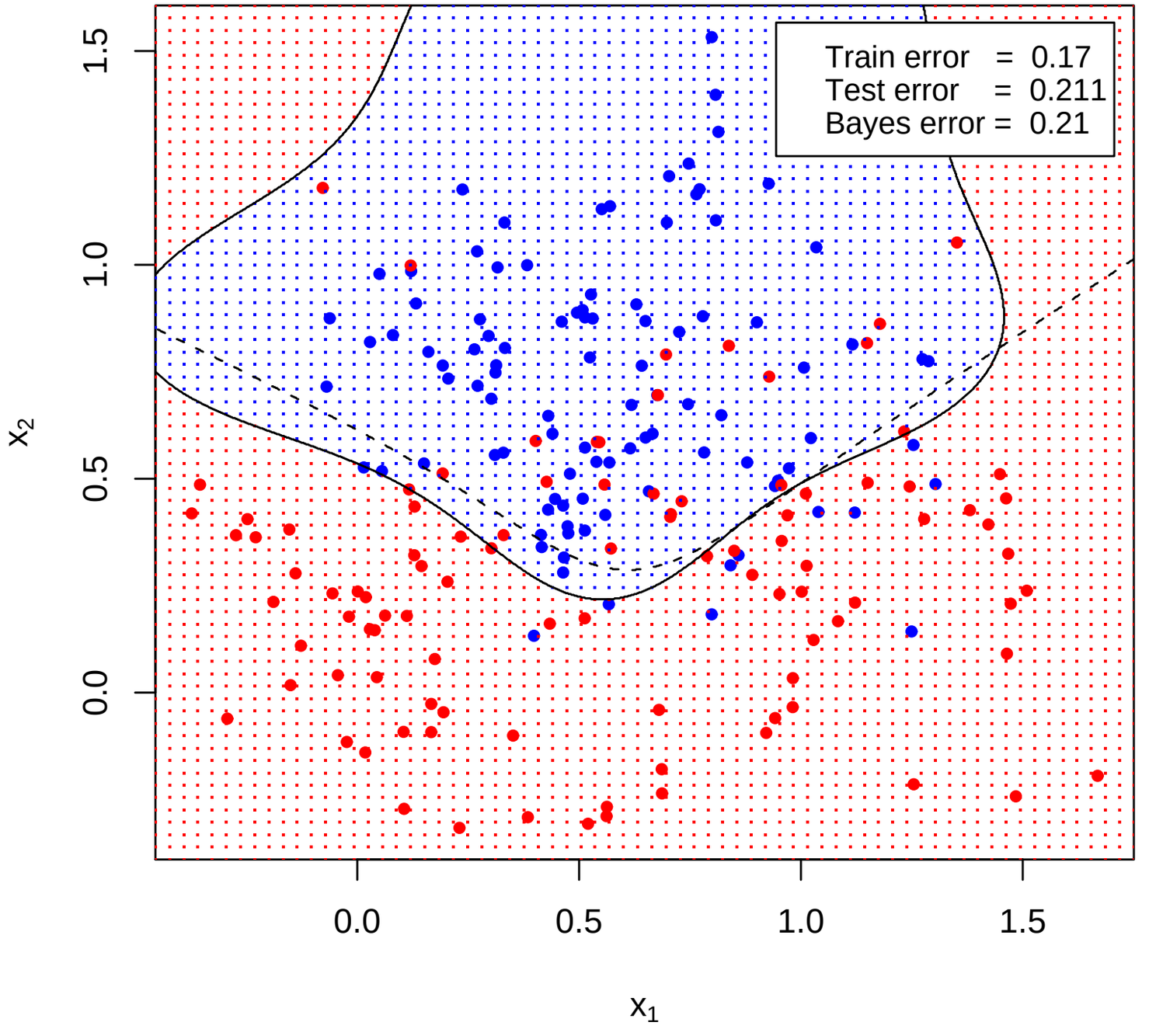
$g(x) = h(x)^\top \beta + \beta_0$ — отступ (margin) объекта x.

SVM эквивалентен минимизации

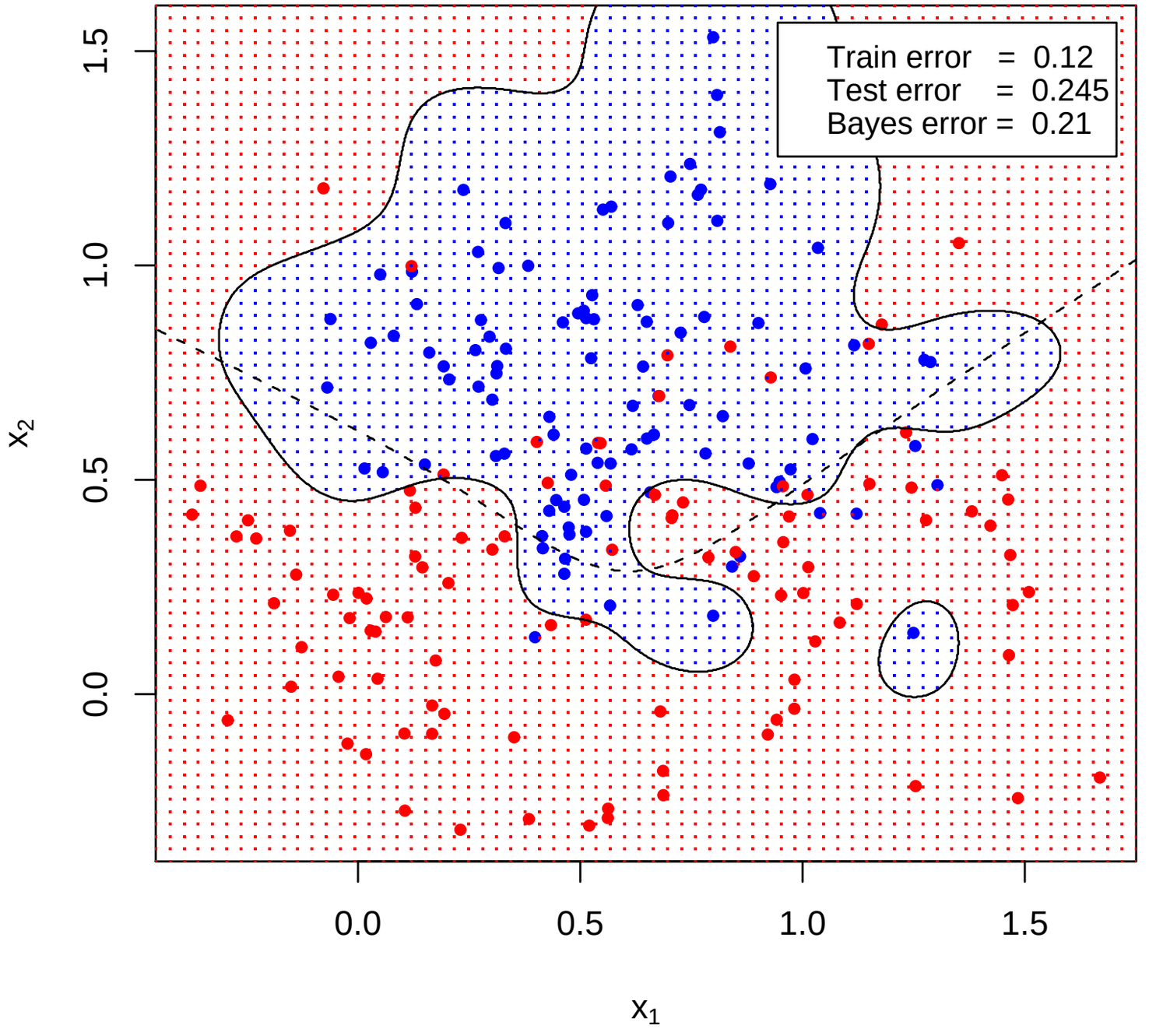
$$\sum_{i=1}^{N} \Big[1 - y^{(i)} \Big(h(x^{(i)})^\top \beta + \beta_0\Big)\Big]_+ + \frac{1}{2\gamma}\|\beta\|^2.$$

Целевая функция имеет вид «потери + штраф».

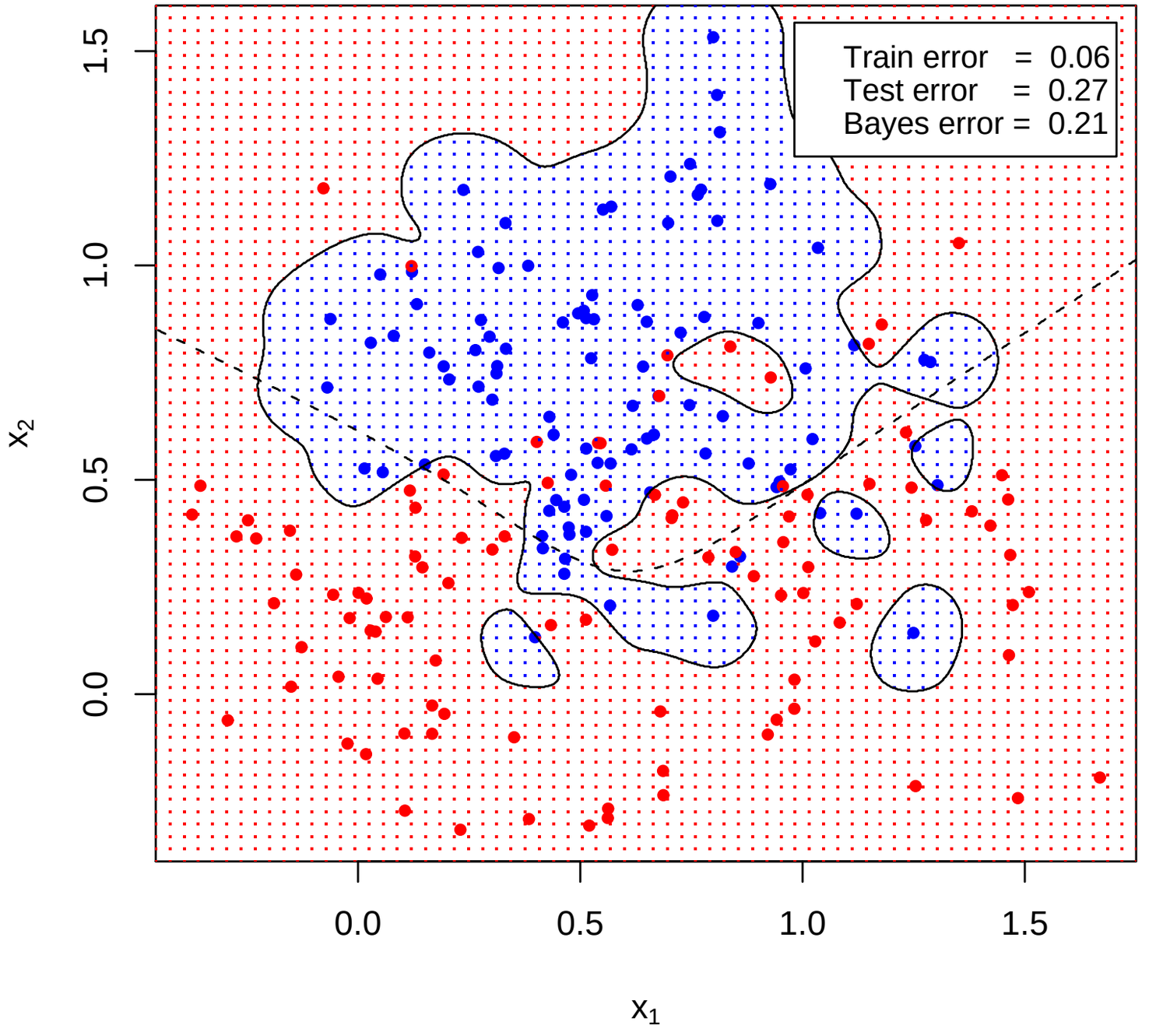
SVM радиальное ядро, $\gamma = 1/2$


Train error = 0.17
Test error = 0.211
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

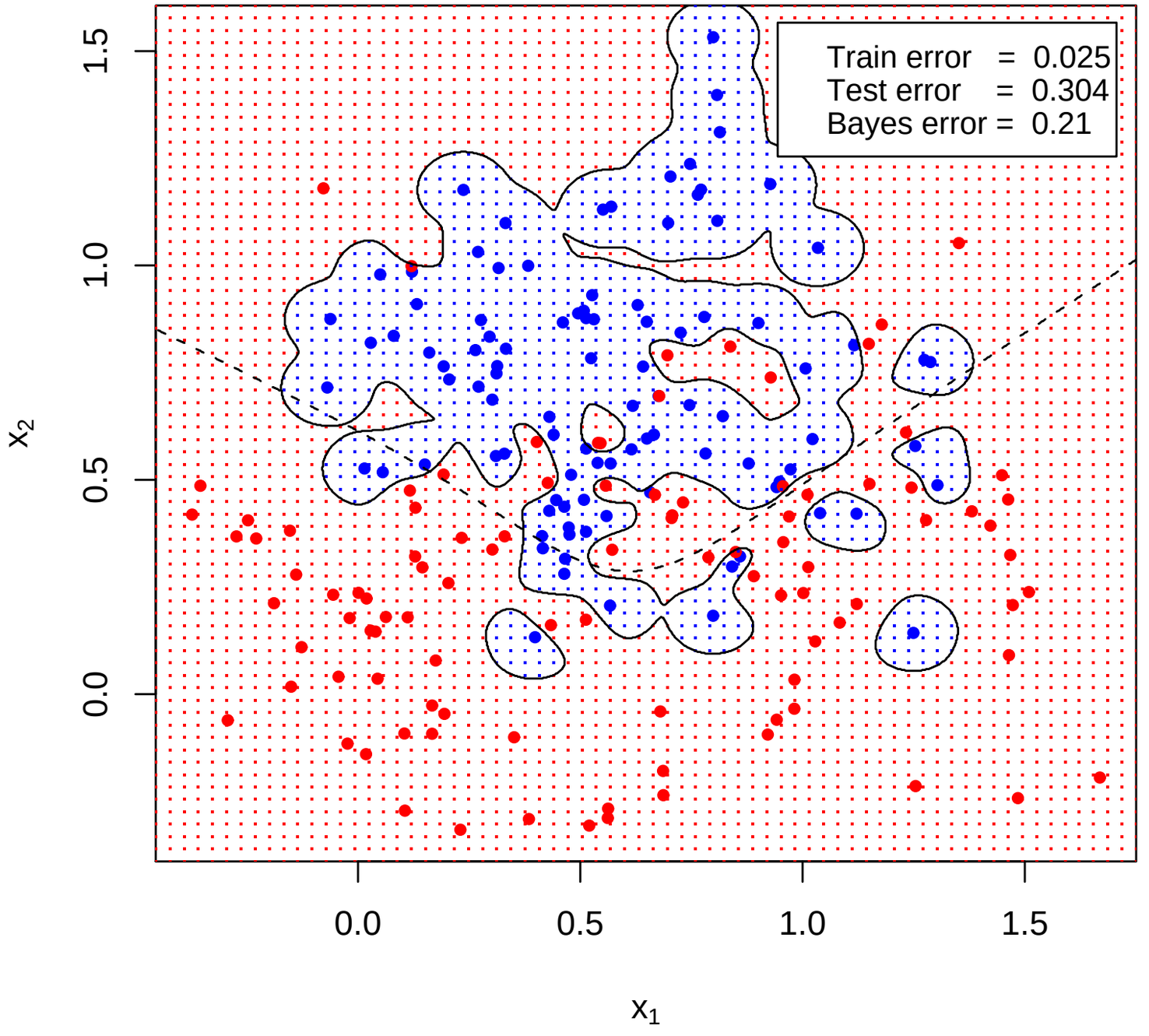
SVM радиальное ядро, $\gamma = 5$


Train error = 0.12
Test error = 0.245
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

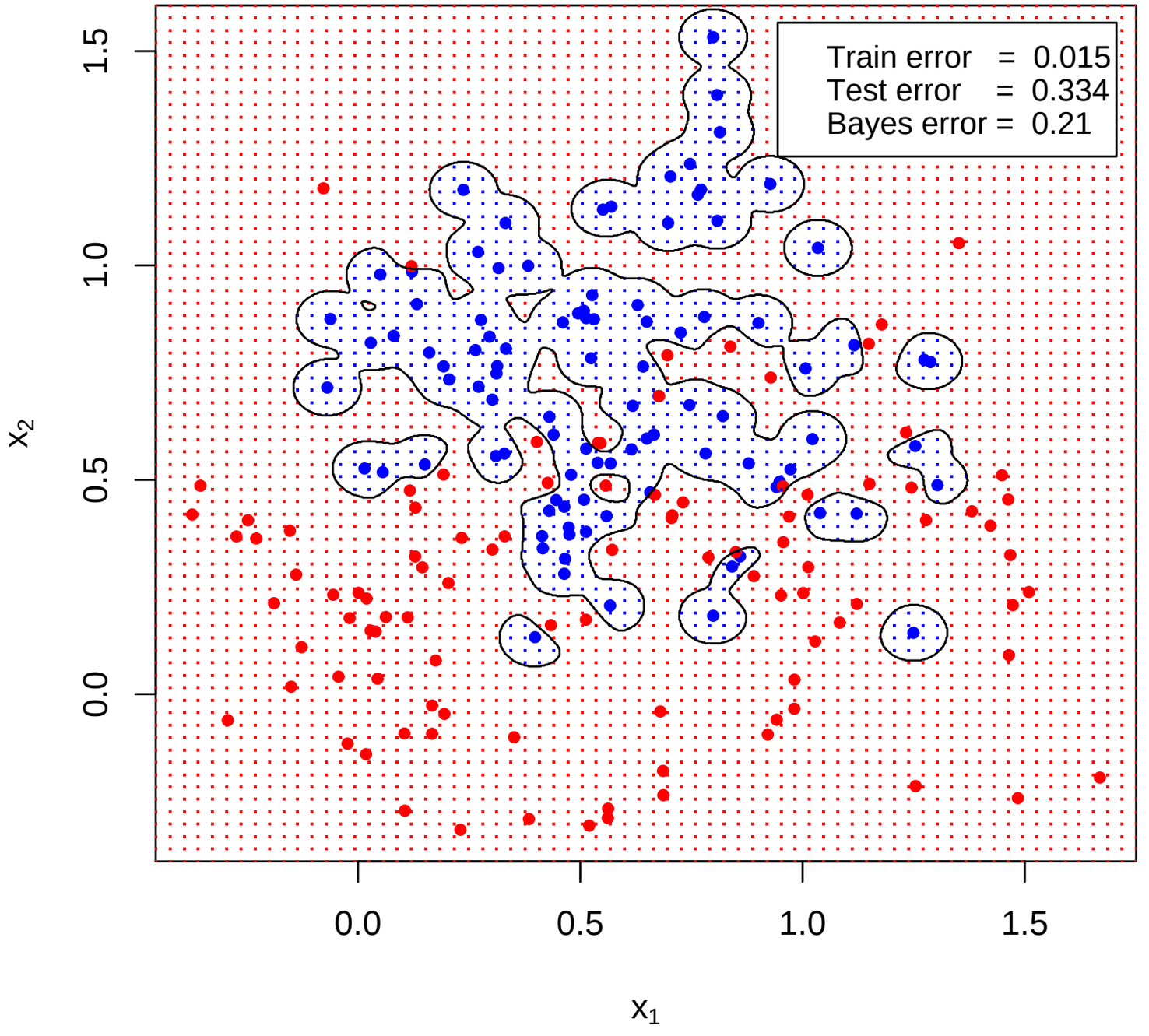
SVM радиальное ядро, $\gamma = 20$


Train error = 0.06
Test error = 0.27
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

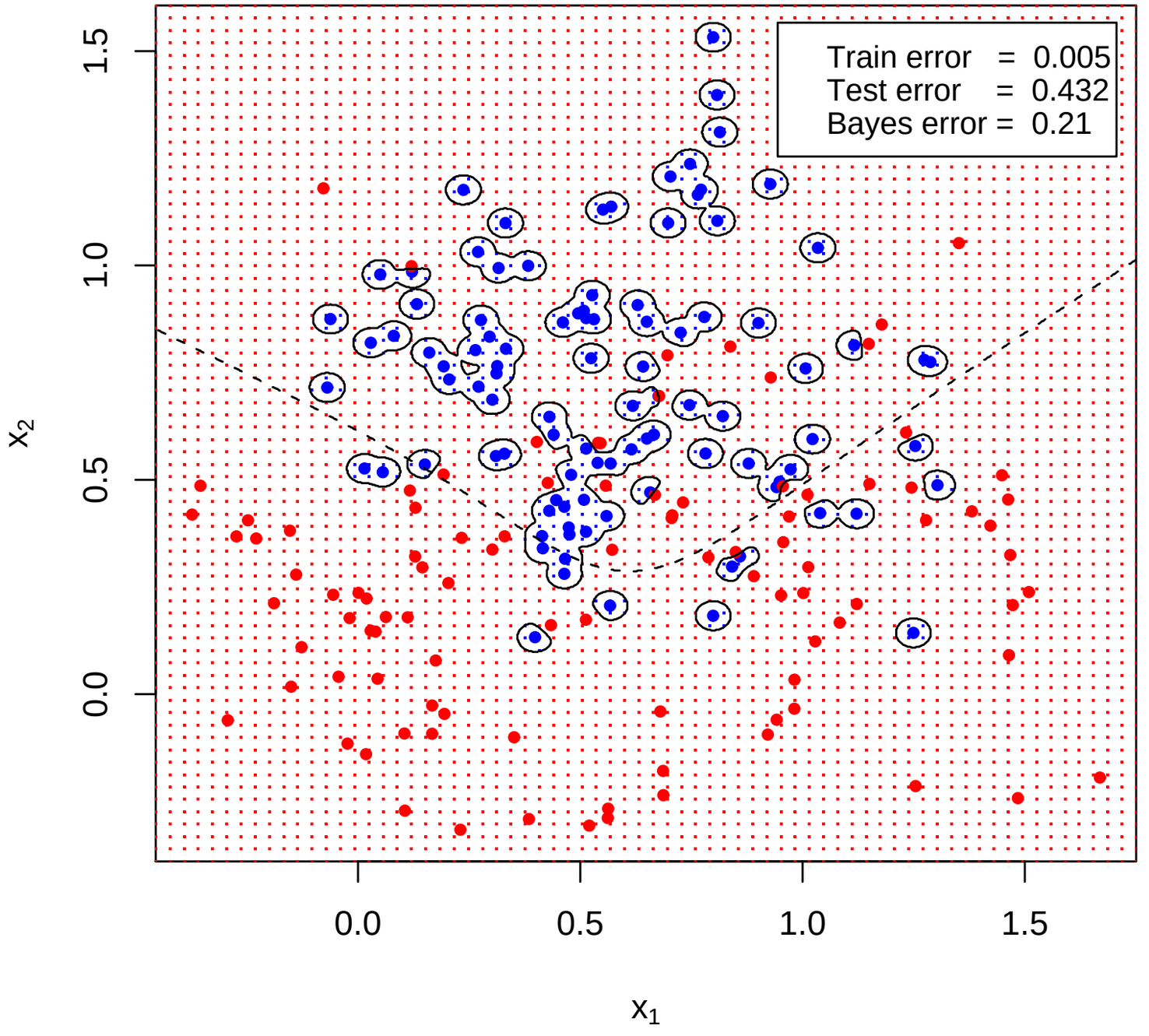
SVM радиальное ядро, $\gamma = 50$


Train error = 0.025
Test error = 0.304
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

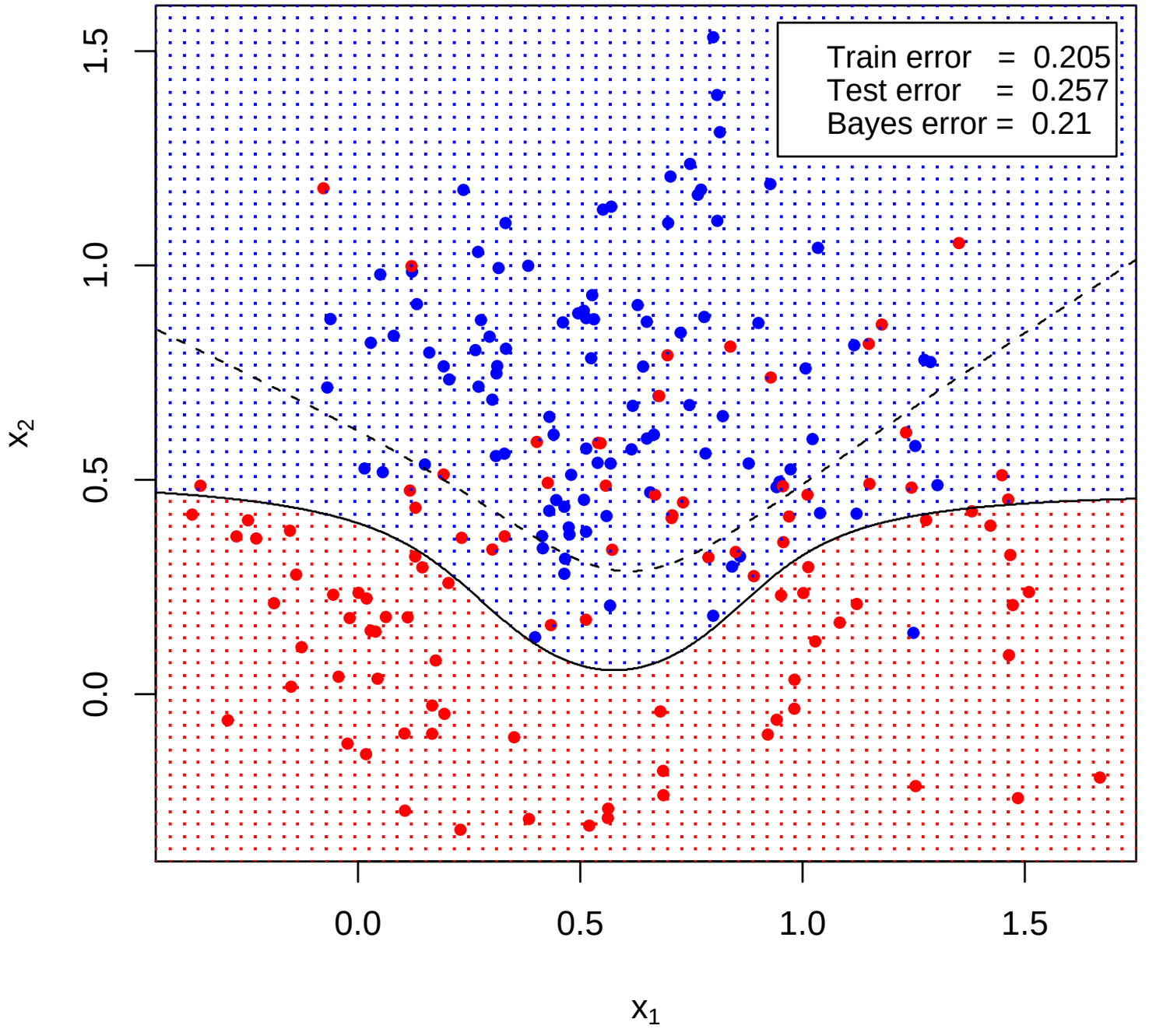
SVM радиальное ядро, $\gamma = 100$


Train error = 0.015
Test error = 0.334
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

SVM радиальное ядро, $\gamma = 500$


Train error = 0.005
Test error = 0.432
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

SVM полиномиальное ядро (полином 3-й степени)

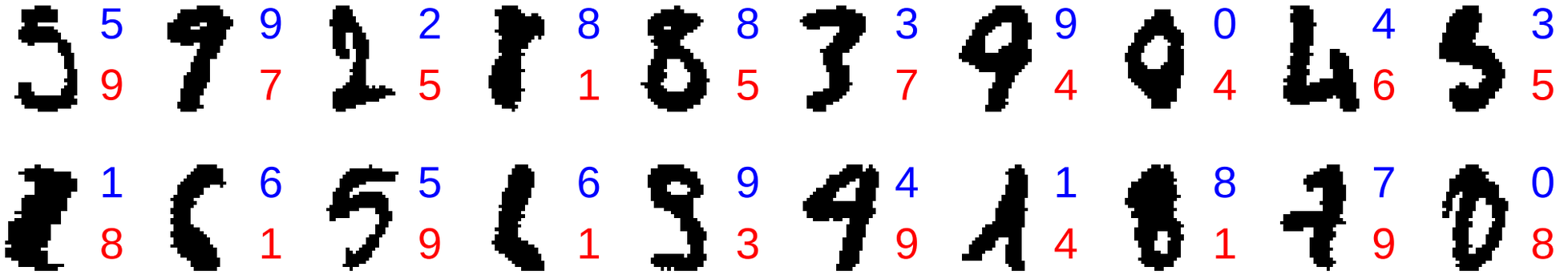

Train error = 0.205
Test error = 0.257
Bayes error = 0.21
0.0
0.5
1.0
1.5
x_1
x_2

Задача классификации рукописных цифр. Выборка размера 1934 была случайным образом разбита на две группы по 967 объектов в каждой. $\gamma = 1/1024$

Ошибки на обучающей и тестовой выборках приведена в следующей таблице.

Ядро	*Ошибка*	
	на обучающей выборке	*на тестовой выборке*
Линейное	0	0.021
Радиальное	0.011	0.030
Полином 3 степени	0.090	0.094
Сигмоидальное	0.131	0.125

Все случаи неправильной классификации цифр из тестовой выборки в случае линейного ядра.



12.4. SVM и восстановление регрессии

Как SVM можно адаптировать к решению задачи восстановления регрессии?

Сначала рассмотрим линейную регрессионную модель

$$f(x) = \beta^\top x + \beta_0.$$

Для восстановления β, β_0 рассмотрим задачу минимизации функции

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V\left(y^{(i)} - f(x^{(i)})\right) + \frac{\alpha}{2}\|\beta\|^2,$$

$$V(t) = V_\varepsilon(t) = \begin{cases} 0, & \text{если } |t| < \varepsilon, \\ |t| - \varepsilon & \text{в противном случае.} \end{cases}$$

$V_\varepsilon(t)$ — функция «ε-нечувствительности», игнорирующая ошибки, меньшие ε.

Можно провести аналогию с SVM-классификатором, в которой точки, расположенные далеко от разделяющей полосы (с «правильной» стороны) не рассматриваются при построении классификатора.

В случае с регрессией такую роль играют точки с маленькой ошибкой $|y^{(i)} - f(x^{(i)})|$.

Можно показать, что решение $\widehat{\beta}$, $\widehat{\beta}_0$, минимизирующее функцию $H(\beta, \beta_0)$, можно представить в виде

$$\widehat{\beta} = \sum_{i=1}^{N} (\widehat{\alpha}_i^* - \widehat{\alpha}_i) x^{(i)}, \quad \widehat{f}(x) = \sum_{i=1}^{N} (\widehat{\alpha}_i^* - \widehat{\alpha}_i) \left\langle x, x^{(i)} \right\rangle + \beta_0,$$

где $\widehat{\alpha}_i$ и $\widehat{\alpha}_i^*$ являются решением следующей задачи квадратичного программирования:

$$\min_{\alpha_i, \alpha_i^*} \left(\varepsilon \sum_{i=1}^{N} (\alpha_i^* + \alpha_i) - \sum_{i=1}^{N} y^{(i)} (\alpha_i^* - \alpha_i) + \frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \left\langle x^{(i)}, x_j \right\rangle \right)$$

при ограничениях

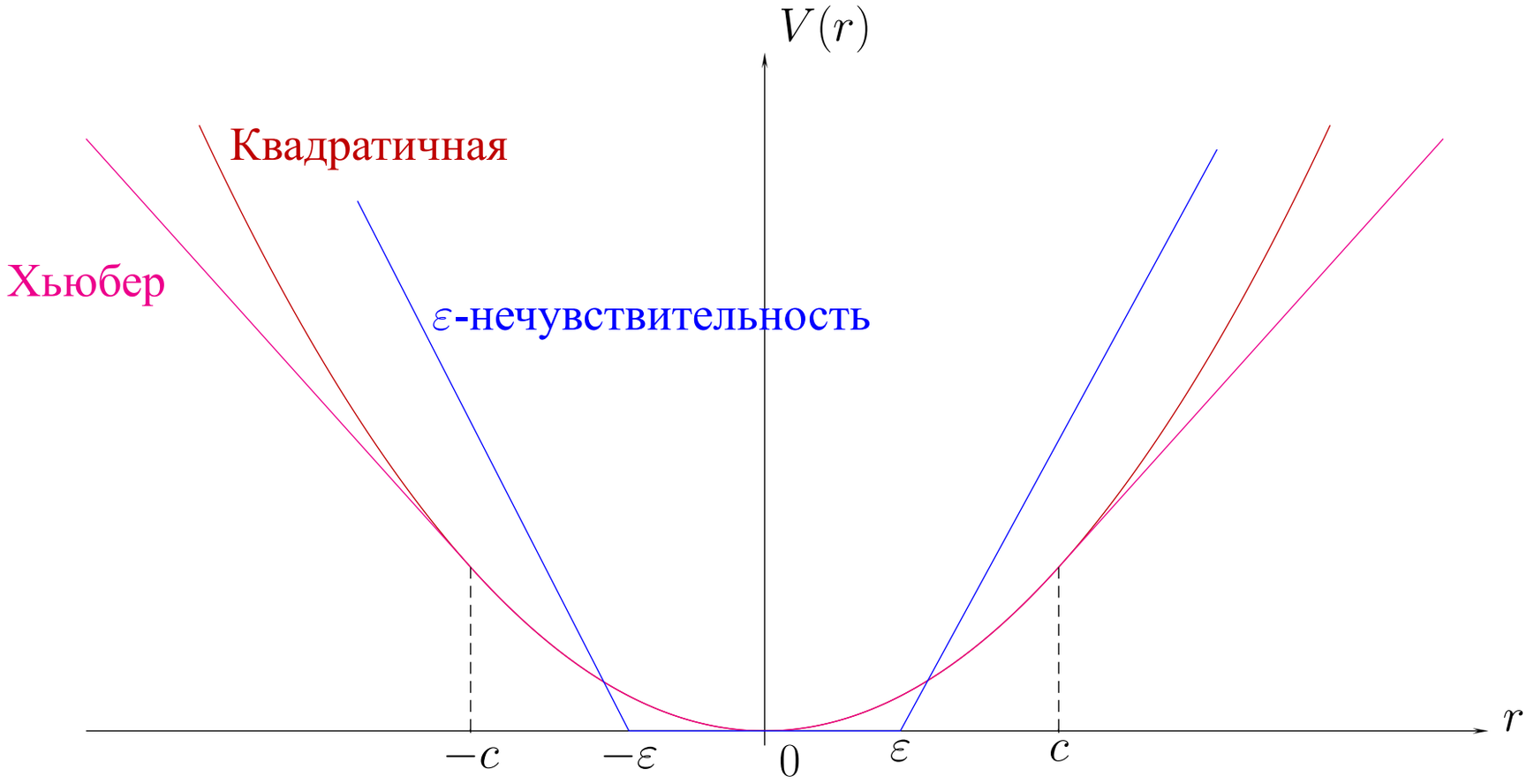
$$0 \leq \alpha_i, \alpha_i^* \leq \frac{1}{\lambda}, \quad \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) = 0, \quad \alpha_i \alpha_i^* = 0.$$

Как и в случае задачи классификации, здесь решение зависит только от скалярных произведений $\left\langle x^{(i)}, x_j \right\rangle$, поэтому мы можем использовать аппарат спрямляющих пространств и ядер.

В качестве функции $V(t)$, можно выбрать другую меру ошибки, например, квадратичную $V(t) = t^2$ или функцию Хьюбера

$$V_H(r) = \begin{cases} r^2/2, & \text{если } |r| < c, \\ c|r| - c^2/2 & \text{в противном случае,} \end{cases}$$

но важно свести все к задаче квадратичного программирования, содержащей только скалярные произведения $\langle x^{(i)}, x_j \rangle$.



12.5. Регрессия и ядра

SVM — не единственная модель, в которой могут использоваться ядра.

Рассмотрим, например, задачу аппроксимации функции $f(x)$ (т. е. задачу восстановления регрессии) при заданных базисных функциях $h_1(x), \ldots, h_m(x)$:

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0.$$

Будем минимизировать регуляризованную функцию потерь

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y^{(i)} - f(x^{(i)})) + \frac{\lambda}{2}\|\beta\|^2.$$

Решение задачи минимизации имеет вид

$$\widehat{f}(x) = \sum_{m=1}^{M} \widehat{\beta}_m h_m(x) + \widehat{\beta}_0 = \sum_{i=1}^{N} \widehat{\alpha}_i K(x, x^{(i)}),$$

где

$$K(x, x') = \sum_{m=1}^{M} h_m(x) h_m(x').$$

Пусть, например, $V(r) = r^2$.

Пусть $\mathbf{H}$ — $(N \times M)$-матрица, в которой (i, m)-й элемент есть $h_m(x^{(i)})$.

Предположим, что $M > N$ и M велико.

Для простоты будем предполагать, что $\beta_0 = 0$.

Тогда

$$H(\beta) = (\mathbf{y} - \mathbf{H}\beta)^\top (\mathbf{y} - \mathbf{H}\beta) + \lambda \|\beta\|^2.$$

Минимизирует функцию $H(\beta)$ вектор $\widehat{\beta}$, который можно определить из условий

$$-\mathbf{H}^\top (\mathbf{y} - \mathbf{H}\widehat{\beta}) + \lambda \widehat{\beta} = 0.$$

После очевидных преобразований получаем

$$\mathbf{H}\widehat{\beta} = (\mathbf{H}\mathbf{H}^\top + \lambda I)^{-1} \mathbf{H}\mathbf{H}^\top \mathbf{y}.$$

Матрица $\mathbf{H}\mathbf{H}^\top$ размера $N \times N$ содержит скалярные произведения для всех пар векторов $\left\langle x^{(i)}, x_j \right\rangle$.

Таким образом, $(\mathbf{H}\mathbf{H}^\top)_{ij} = K(x^{(i)}, x_j)$.

Легко показать, что

$$\widehat{f}(x) = h(x)^\top \beta = \sum_{i=1}^{N} \widehat{\alpha}_i K(x, x^{(i)}),$$

где $\widehat{\alpha} = (\mathbf{H}\mathbf{H}^\top + \mathbf{I})^{-1}\mathbf{y}$.

Как и в SVM, нет необходимости вычислять M значений функций $h_1(x), \ldots, h_M(x)$ (и даже определять сами эти функции).

Достаточно вычислить только значения $K(x^{(i)}, x_j)$.

Удачный выбор функций h_m (например, если в качестве них выбраны собственные функции ядра) позволяет вычислить все эти значения за время $N^2/2$, а не $N^2 M$, как при прямом умножении матриц.