Федеральное государственное автономное образовательное учреждение высшего образования
Санкт-Петербургский Политехнический Университет Петра Великого
Высшая школа прикладной математики и вычислительной физики

Курсовая работа

Построение SV-классификатора (SVM)

Преподаватель:	Павлова Людмила Владимировна
Выполнил студент гр. 5030102/10401:	Прохоров Артём Дмитриевич
Дата:	27 мая 2025 г.

Санкт-Петербург
2025

Содержание

1 Постановка задачи

I. SV-подход к решению задачи бинарной классификации:

Необходимо изложить основные идеи и результаты.
(Для **soft margin SVM** – требуется вывод двойственной задачи L_D)

II. Практическая часть

1. **Линейно разделимые данные:**
   - Подготовить тренировочную выборку объёмом около 100 объектов, представляющих точки в двумерном пространстве, которые допускают линейное разделение.
   - Решить соответствующую задачу квадратичного программирования в формулировке Вольфа (например, в MATLAB можно использовать функцию quadprog).
   - Определить опорные векторы (SV) как те, у которых коэффициенты Лагранжа $a_i \geq tol$, где $tol = 1e - 3$.
   - Представить:
     - полученные результаты,
     - графическую интерпретацию исходных данных и результатов:
     - разделяющая гиперплоскость и границы,
     - опорные векторы (SV) и пограничные опорные векторы (BSV), если таковые имеются, и их количество,
     - ширина зазора (margin, M).
2. **Нарушенные (зашумленные) линейные данные:**
   - Внести несколько ошибок в обучающую выборку.
   - Построить SV-машину для изменённой (зашумленной) выборки.
   - Исследовать влияние штрафного параметра C на ширину полосы (margin, M).
   - Вычислить и представить значения M для различных C, например: C = 0.1, 1, 10, 100, 1000 и т.д.
   - Построить соответствующие графики и визуализации, аналогично п.1.
3. **Нелинейно разделимые данные:**
   - Подготовить выборку объёмом около 100 объектов, которая не допускает линейного разделения, но разделима нелинейно.
   - Повторить п.1, заменив скалярное произведение на гауссово (радиальное) ядро или другое (см. формулы из лекции).
   - Изучить, как результаты зависят от выбора параметра σ^2 (или других параметров других ядер).
4. **Нарушенные нелинейные данные:**
   - Внести ошибки в выборку из п.3.

- Провести аналогичные исследования, как в п.2, для случая нелинейного разделения:
- влияние С,
- ширина зазора М,
- визуализации.

III. Работа с реальными данными

- Выбрать подходящий по объему датасет из репозитория machine-learning-databases.
- Выполнить обучение SV-классификатора.
- Построить визуализацию.
- Сделать выводы.

2 Алгоритм

1. **Постановка задачи бинарной классификации**

   - **Данные:** пары (x_i, y_i), где $x_i \in R^n, y_i \in \{\pm 1\}$
   - **Цель:** найти функцию $f(x) : R^n \to \{\pm 1\}$, минимизирующую эмпирический риск:

   $$R_{emp}[f] = \frac{1}{l} \sum_{i=1}^{l} L(f(x_i), y_i),$$

   где L — функция потерь (например, $L(f(x), y) = \frac{1}{2}||y - f(x)\rangle$).

2. **Линейный SVM для линейно разделимых данных (Hard Margin)**

   - **Гиперплоскость:** $f(x) = (w * x) + b$.
   - **Условия:**

   $$y_i((w * x_i) + b) \geq 1, \forall i.$$

   - **Оптимизация:** Минимизация $\frac{1}{2}|w|^2$ при заданных ограничениях.
   - **Двойственная задача (в формулировке Вульфа):**

   $$\max_{\alpha} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j (x_i * x_j),$$

   с условиями $\sum_{i=1}^{l} \alpha_i y_i = 0$ и $\alpha_i \geq 0$.

3. **Линейный SVM для линейно неразделимых данных (Soft Margin)**

   - **Добавление переменных** ξ_i (нарушений границы):

   $$y_i((w * x_i)b) \geq 1 - \xi_i, \xi_i \geq 0.$$

- **Оптимизация:**

$$\min_{w,b} \frac{1}{2}|w|^2 + C\sum_{i=1}^{l} \xi_i.$$

- **Двойственная задача (Вульфа)** аналогична, но с ограничением $(0 \leq \alpha_i \leq C)$.

4. **Нелинейный SVM**

- **Ядерные функции:** Преобразование $\varphi : X \to H$, где H - пространство признаков.
- **Примеры ядер:**
  - Линейное: $k(x, x') = (x * x')$
  - Гауссово (RBF): $k(x, x') = e^{-|x-x'|^2/2\sigma^2}$,
  - Полиномиальное: $k(x, x') = (\gamma(x * x') + r)^p$.
- **Решающее правило:**

$$f(x) = sgn(\sum_{i=1}^{l} \alpha_i y_i k(x_i, x) + b).$$

5. **Оценка качества классификации**

- LOO-оценка (Leave-One-Out):

$$R_{l00}[f] = \frac{1}{l}\sum_{j=1}^{l} L(f^j(x_j), y_j),$$

  где f^j обучается на данных без j-го наблюдения.
- Кросс-валидация: Разделение данных на k частей и оценка на каждой части.

6. **Преимущества SVM**

- Эффективность в высокоразмерных пространствах.
- Использование ядер для нелинейных задач.
- Разреженность решения (зависит только от опорных векторов)

7. **Практические шаги**

7.1 Подготовка данных (масштабирование, обработка выбросов).

7.2 Выбор ядерной функции.

7.3 Настройка гиперпараметров (например,C, σ).

7.4 Обучение и оценка модели.

Вывод двойственной задачи для Soft Margin SVM

1. **Исходная задача**

Для линейно неразделимых данных вводим переменные ослабления $\xi_i \geq 0$:

$$\min_{w,b,\xi} \frac{1}{2}|w|^2 + C\sum_{i=1}^{l} \xi_i,$$

с ограничениями:

$$y_i((w * x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0, \forall i.$$

2. **Функция Лагранжа**

   Добавляем множители Лагранжа:

   - $\alpha_i \geq 0$ для ограничений $y_i(w * x_i + b) \geq 1 - \xi_i$,
   - $\mu_i \geq 0$ для условий $\xi_i \geq 0$.

   Функция Лагранжа:

$$L(w, b, \xi, \alpha, \mu) = \frac{1}{2}|w|^2 + C\sum_{i=1}^{l} \xi_i - \sum_{i=1}^{l} \alpha_i[y_i(w * x_i + b) - 1 + \xi_i] - \sum_{i=1}^{l} \mu_i \xi_i.$$

3. **Условия оптимальности (KKT)**

   Приравниваем производные к нулю:

   3.1 По w:
   $\nabla_w L = w - \sum_{i=1}^{l} \alpha_i y_i x_i = 0 \implies w = \sum_{i=1}^{l} \alpha_i y_i x_i.$

   3.2 По b:
   $\frac{\partial L}{\partial b} = -\sum_{i=1}^{l} \alpha_i y_i = 0 \implies \sum_{i=1}^{l} \alpha_i y_i = 0.$

   3.3 По ξ_i :
   $\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \mu_i = 0 \implies \alpha_i = C - \mu_i.$
   Так как $\mu_i \geq 0$, то $a_i \leq C$.

4. **Подстановка в функцию Лагранжа**

   Подставляем $w = \sum \alpha_i y_i x_i$ и упрощаем:

$$L = \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i\alpha_j y_i y_j (x_i * x_j) + C\sum_{i=1}^{l} \xi_i - \sum_{i,j=1}^{l} \alpha_i\alpha_j y_i y_j (x_i * x_j) - b\sum_{i=1}^{l} \alpha_i y_i + \sum_{i=1}^{l} \alpha_i - \sum_{i=1}^{l} \alpha_i \xi_i - \sum_{i=1}^{l} \mu_i \xi_i.$$

   Учитывая $\sum \alpha_i y_i = 0$ и $\mu_i = C - \alpha_i$, сокращаем:

$$L_D(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i\alpha_j y_i y_j (x_i * x_j).$$

5. **Двойственная задача**

$$\max_{\alpha} L_D(\alpha) = \sum_{i=1}^{l} \alpha_i \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i\alpha_j y_i y_j (x_i * x_j),$$

   с ограничениями:

$$\sum_{i=1}^{l} \alpha_i y_i = 0, 0 \leq \alpha_i \leq C, \forall i.$$

3 Применение алгоритма

1. **Линейно разделимые данные**

1.1 Сгенерируем двумерные линейно разделимые данные.

1.2 Построим SVM через собственную реализацию:

- вручную сформулируем двойственную задачу в виде задачи квадратичного программирования (QP);
- решим её с помощью cvxopt.

1.3 Визуализируем: данные, разделяющую гиперплоскость, margin, опорные векторы.

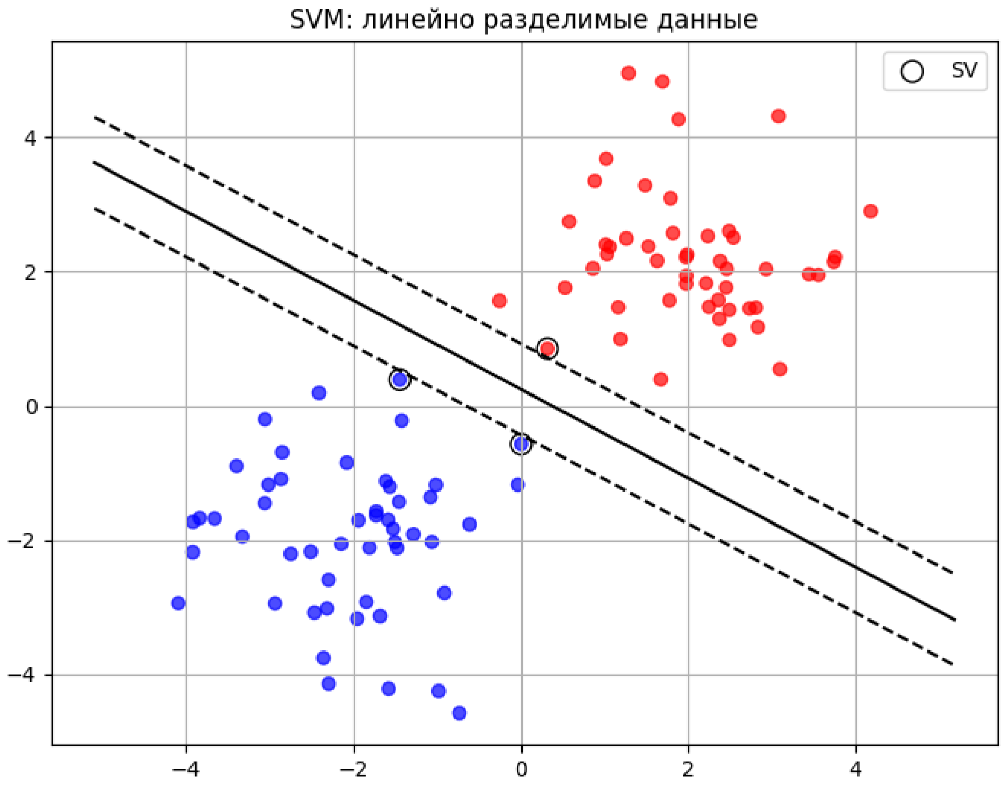


Рис. 1:
Количество опорных векторов: 3
Ширина зазора (margin): 1.3577

2. **Линейно разделимые данные с шумом**

2.1 Добавить **шум (ошибки)** в линейно разделимую выборку.

2.2 Решим **soft-margin SVM** с различными значениями C, используя двойственную формулировку.

2.3 Визуализируем результаты:

- Разделяющую гиперплоскость,
- Опорные и **пограничные опорные векторы** (BSV),
- **Ширину зазора** для разных C.

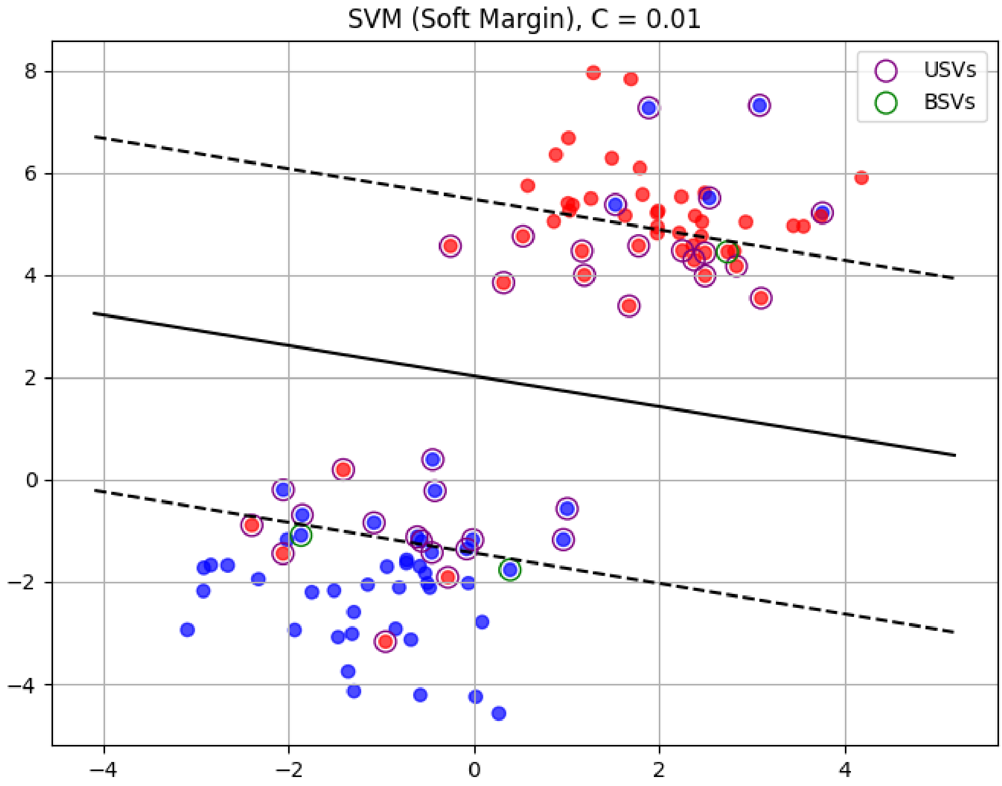


Рис. 2:
C = 0.01, Кол-во SV = 38, BSV = 3, Margin = 6.6270

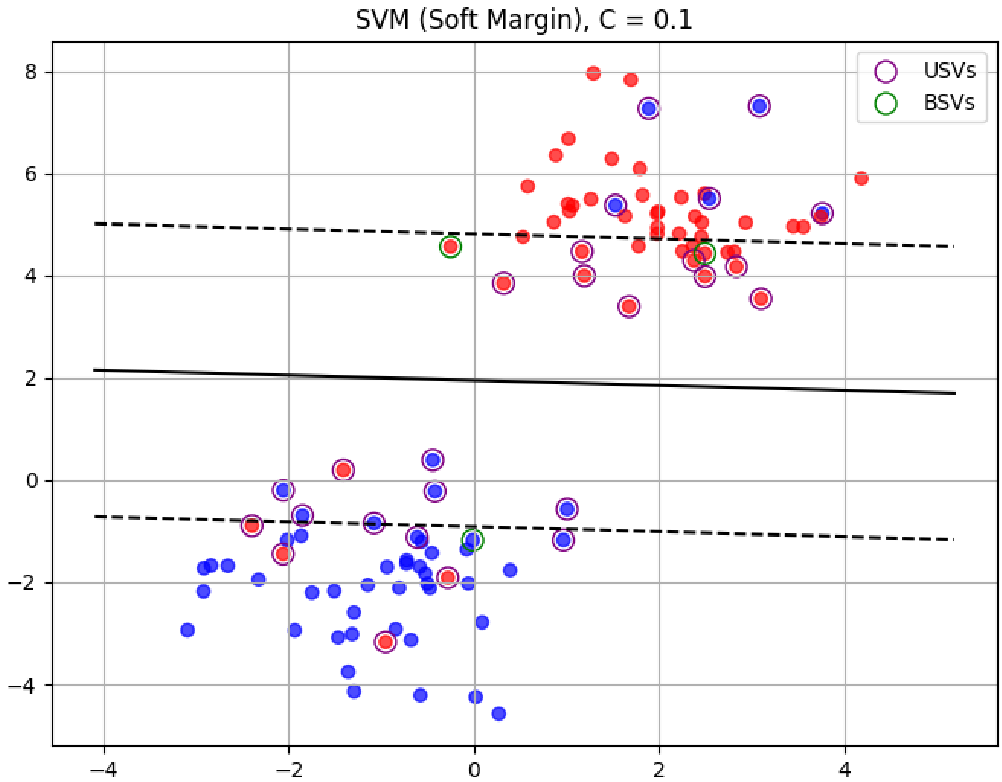


Рис. 3:
C = 0.1, Кол-во SV = 29, BSV = 3, Margin = 5.7241

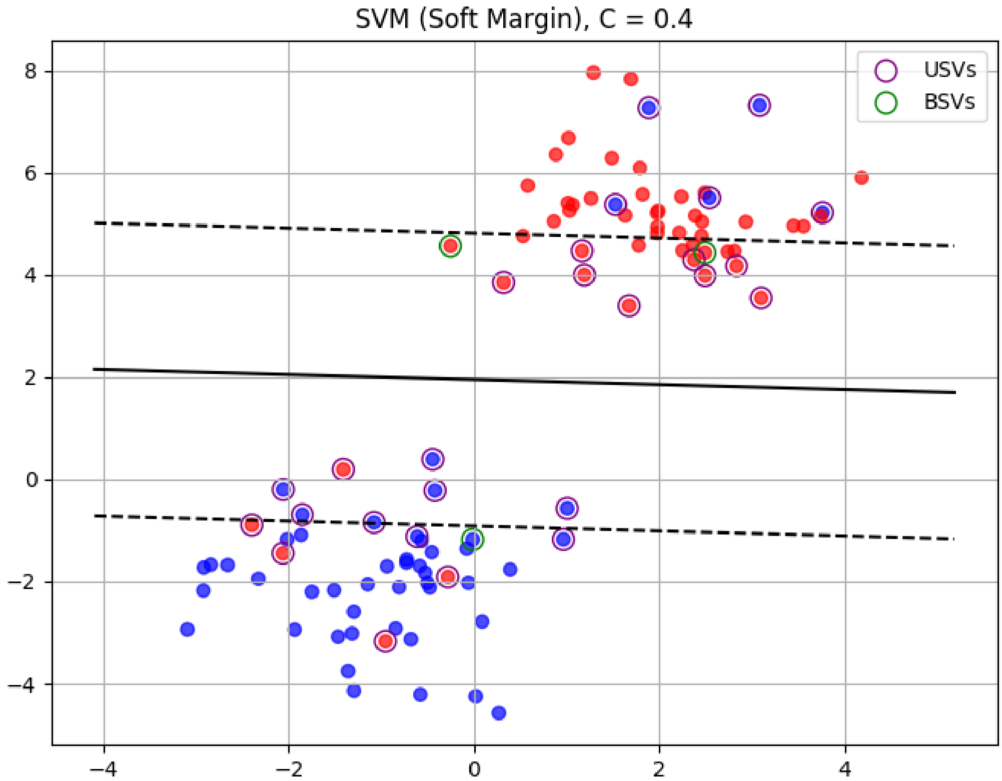


Рис. 4:
C = 0.4, Кол-во SV = 29, BSV = 3, Margin = 5.7239

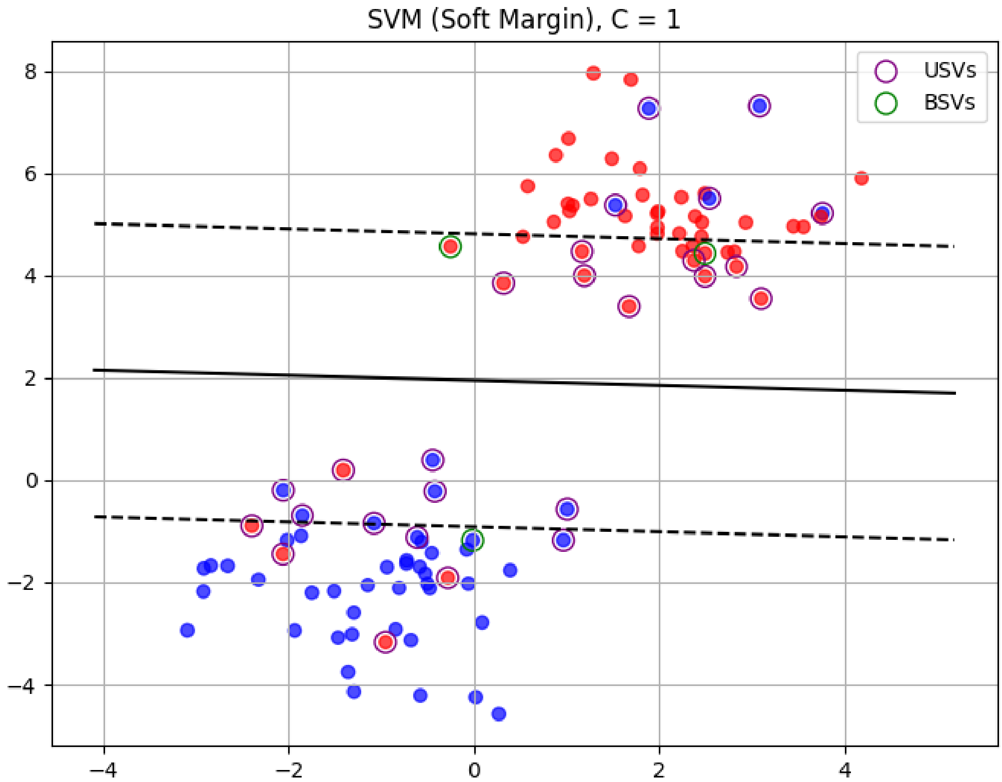


Рис. 5:
C = 1, Кол-во SV = 29, BSV = 3, Margin = 5.7248

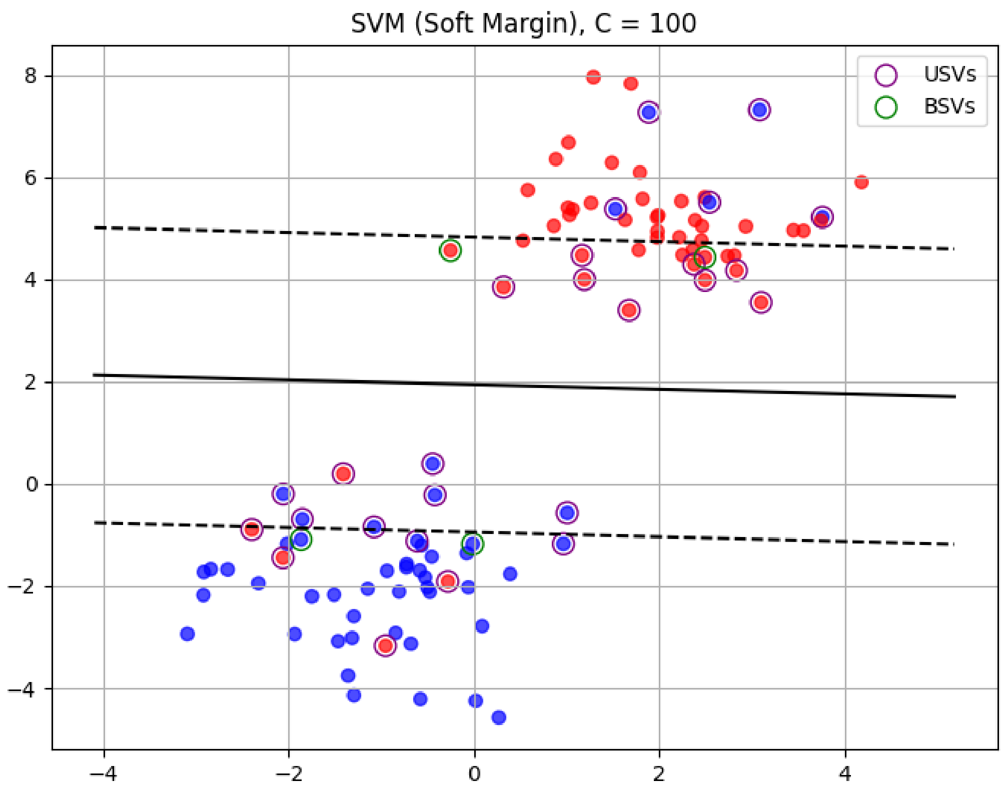


Рис. 6:
C = 100, Кол-во SV = 30, BSV = 4, Margin = 5.7688

3. **Линейно неразделимые данные**

   3.1 Сгенерируем **нелинейно разделимую выборку** (например, кольца).

   3.2 Решим задачу с помощью **ядрового SVM**. Используем ядро **RBF**.

   3.3 Визуализируем:

   - Нелинейную границу,
   - Опорные векторы,
   - Эффект параметра ядра (γ).

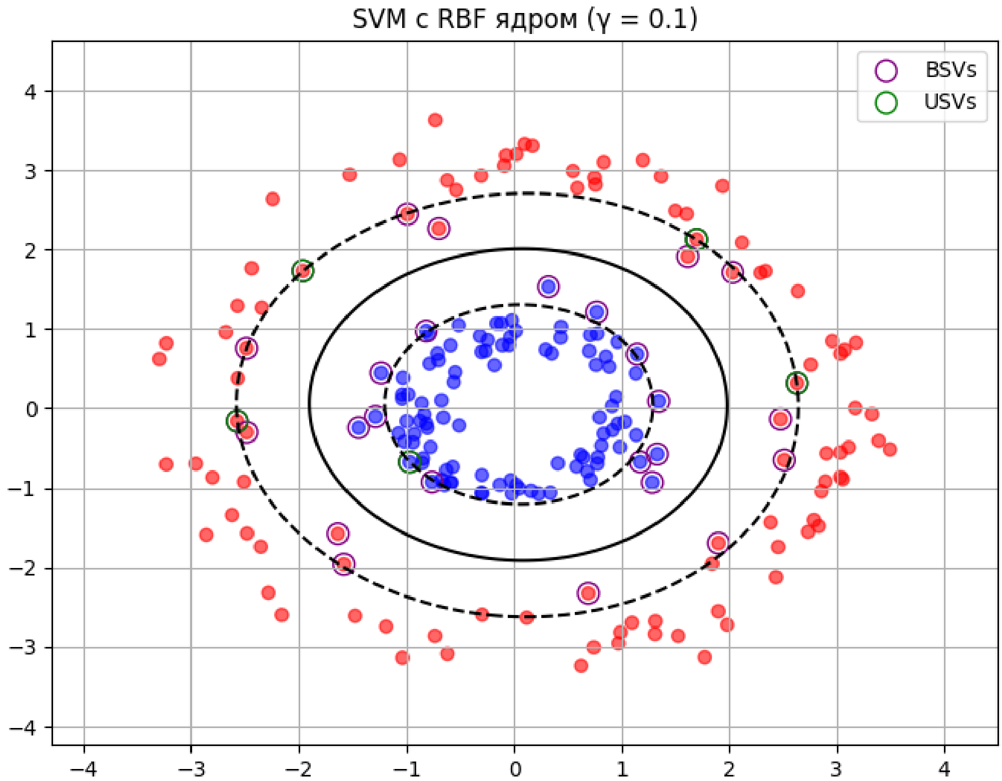


Рис. 7:
Gamma = 0.1 Кол-во опорных векторов: 29 Кол-во BSV: 24 Ширина зазора (margin): 0.4286

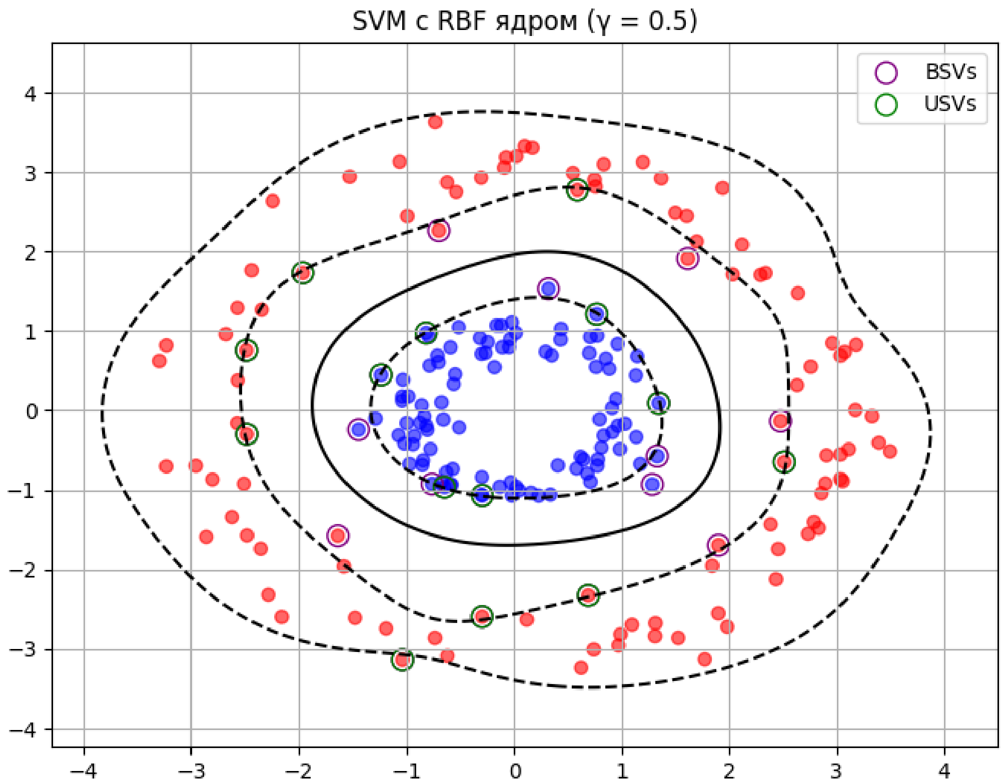


Рис. 8:
Gamma = 0.5 Кол-во опорных векторов: 24 Кол-во BSV: 10 Ширина зазора (margin): 0.5331

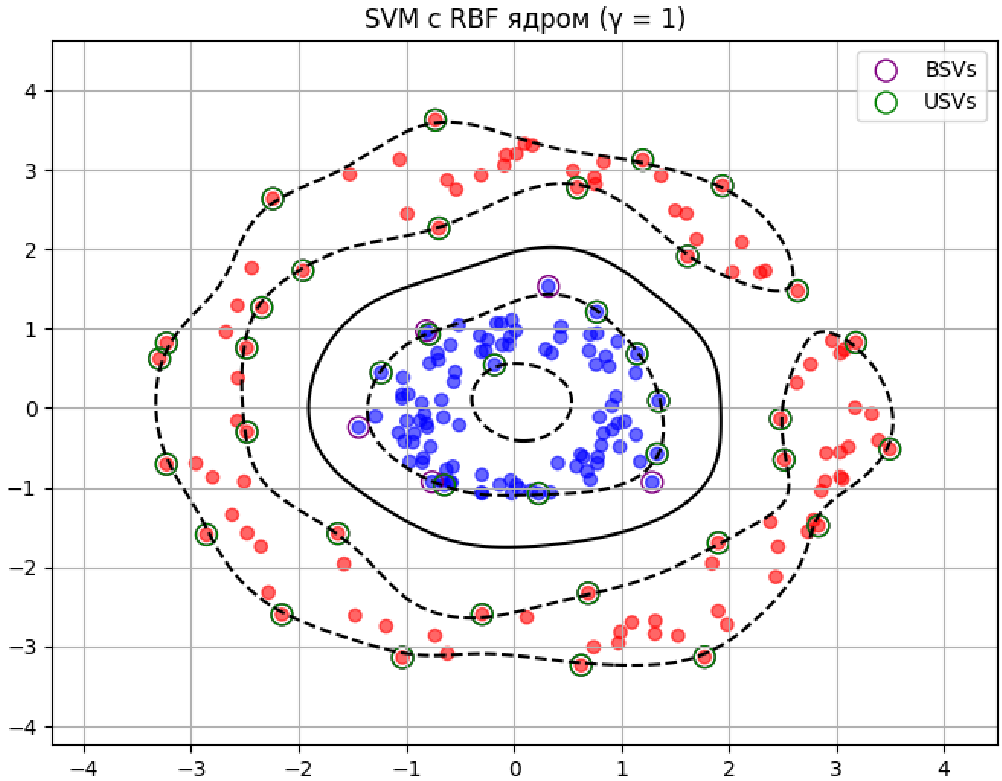


Рис. 9:
Gamma = 1 Кол-во опорных векторов: 43 Кол-во BSV: 5 Ширина зазора (margin): 0.5014

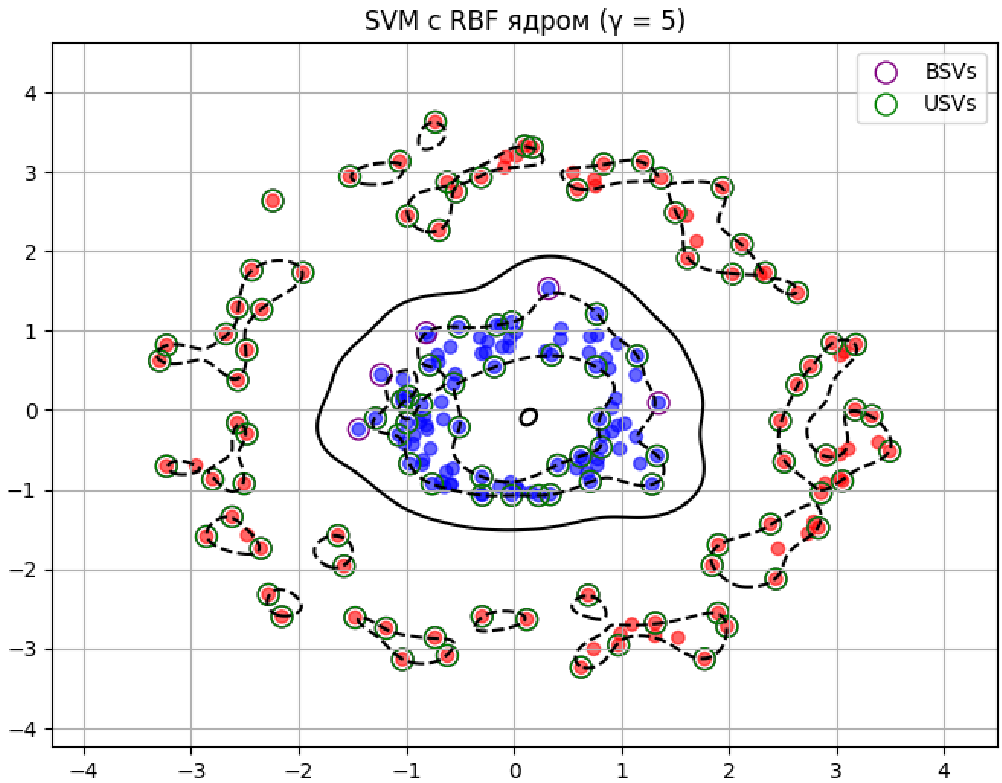


Рис. 10:
Gamma = 5 Кол-во опорных векторов: 110 Кол-во BSV: 5 Ширина зазора (margin): 0.3226

4. **Линейно неразделимые данные с шумом**

4.1 Внесем **шум (ошибки в метки)** в данные из 3.3 (кольца).

4.2 Протестируем **влияние параметров** на:

- Количество опорных векторов,
- Ширину "зазора"M,
- Форму границы.

4.3 Визуализировать результаты.

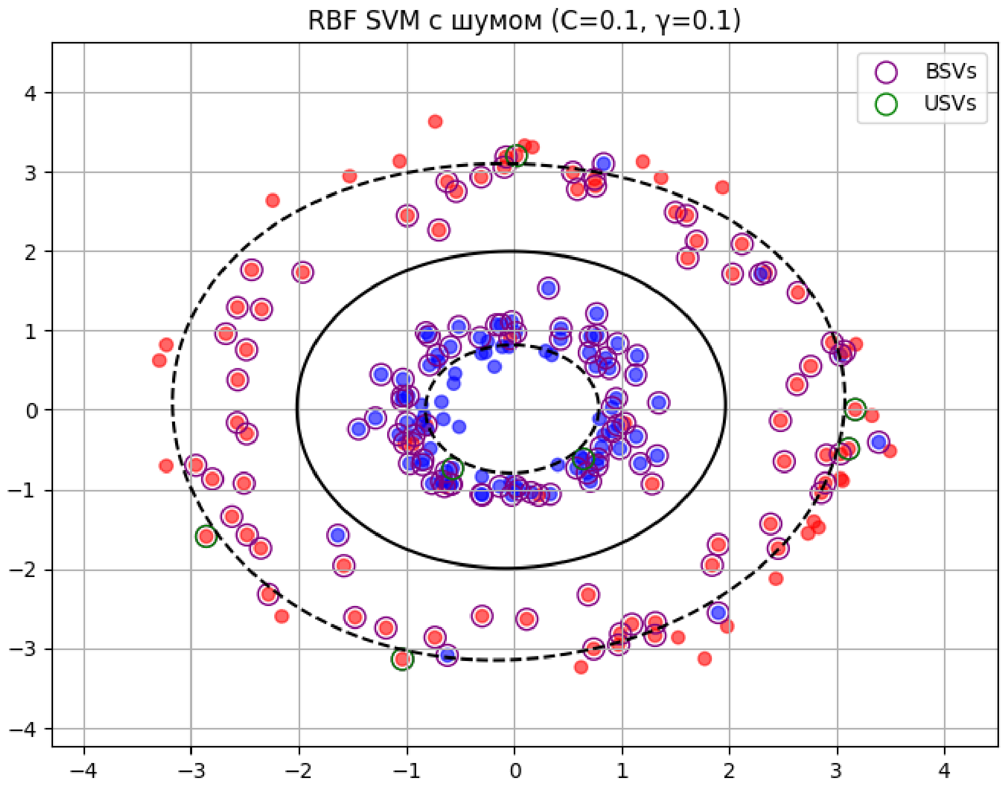


Рис. 11:
C = 0.1 | Кол-во опорных векторов: 149 | BSVs: 142 | Оценка зазора (M): 2.484

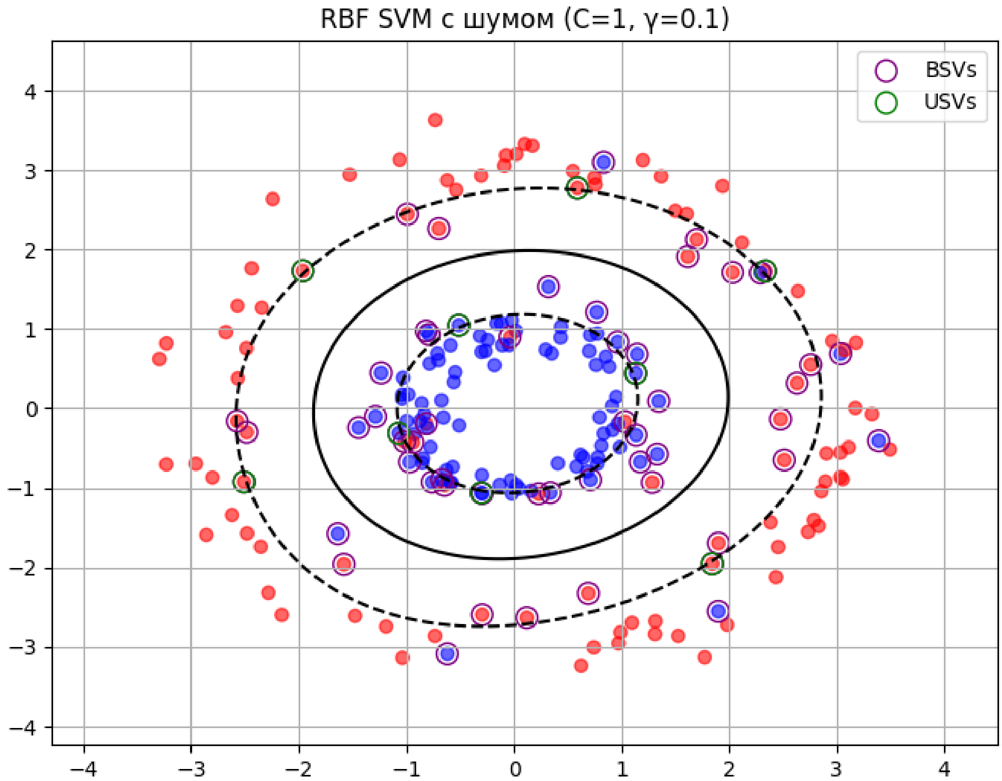


Рис. 12:
C = 1 | Кол-во опорных векторов: 59 | BSVs: 50 | Оценка зазора (M): 2.208

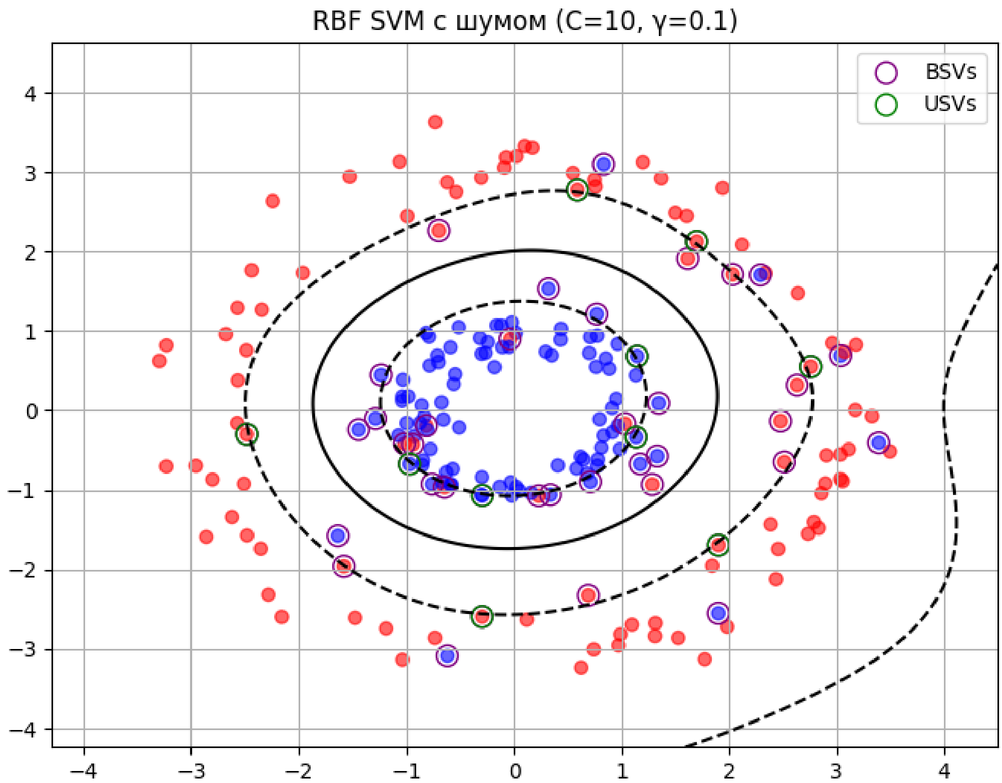


Рис. 13:
C = 10 | Кол-во опорных векторов: 44 | BSVs: 34 | Оценка зазора (M): 2.042

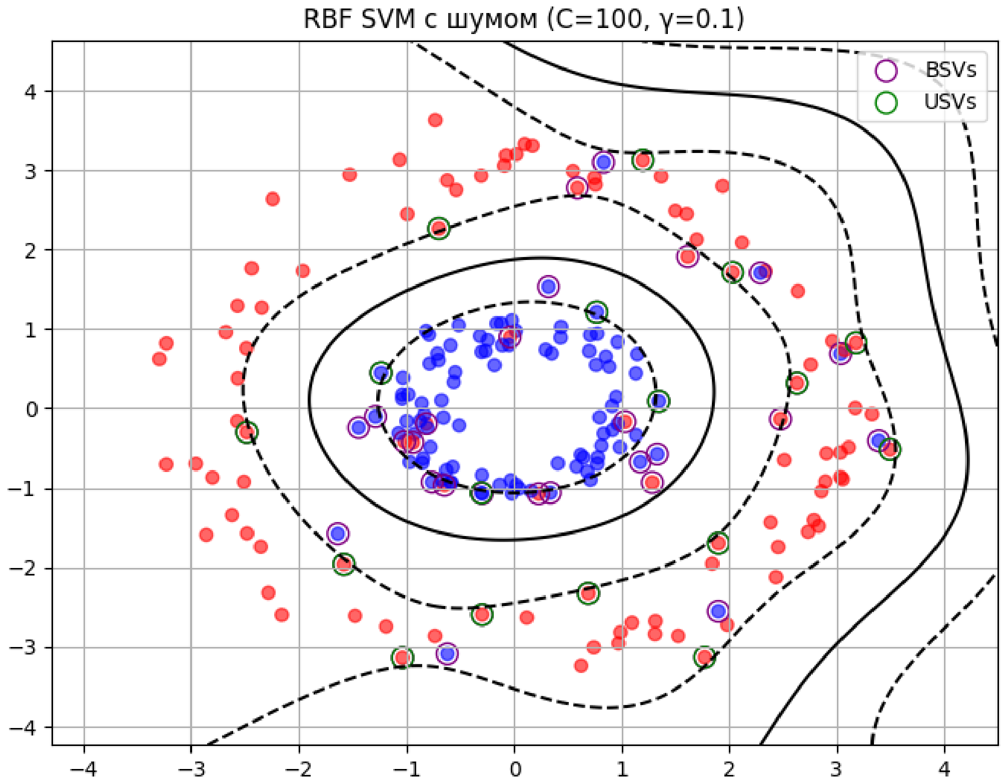


Рис. 14:
C = 100 | Кол-во опорных векторов: 43 | BSVs: 26 | Оценка зазора (M): 1.984

Мы можем видеть, что параметры C и γ являются параметрами, регулирующими сложность моделей. Чем выше значение C, тем сильнее мы штрафуем модель за ошибки, внутри разделяющей плоскости. То есть, при уменьшении значения параметра C мы сильнее учитываем выбросы при разделении данных на классы. С увеличением же γ мы можем строить более сложные разделяющие гиперплоскости, и сильнее подстраиваться под формы расположения данных в пространстве признаков (входных данных).

Наиболее грамотным будет построение модели с помощью жадного или оптимизированного перебора этих параметров для нахождения оптимальных по итоговому качеству.

5. Реальные данные

Для текущего шага будет использоваться датасет Heart. Он содержит информацию о пациентах, например, их возраст, пол, артериальное давление и уровень сахара в крови. Выход - наличие болезней сердца.

5.1 Загрузим реальные данные heart.

5.2 Разделим на тренировочную и валидацонную выборки (80/20).

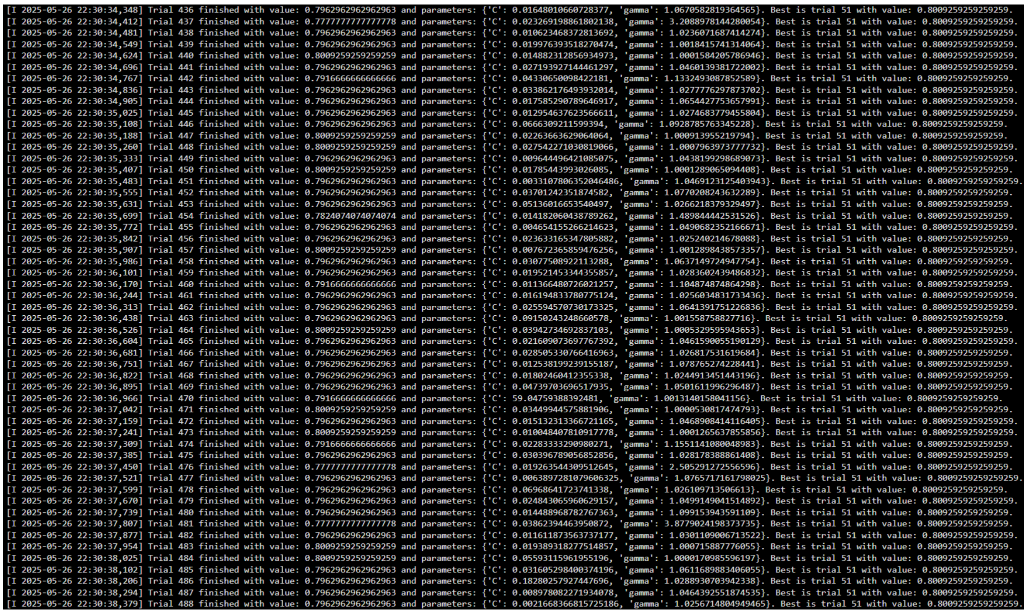
5.3 Стандартизируем их и приведем к виду, удобному для обучения модели.

5.4 Решим задачу с помощью ядрового SVM. Используем ядро RBF.

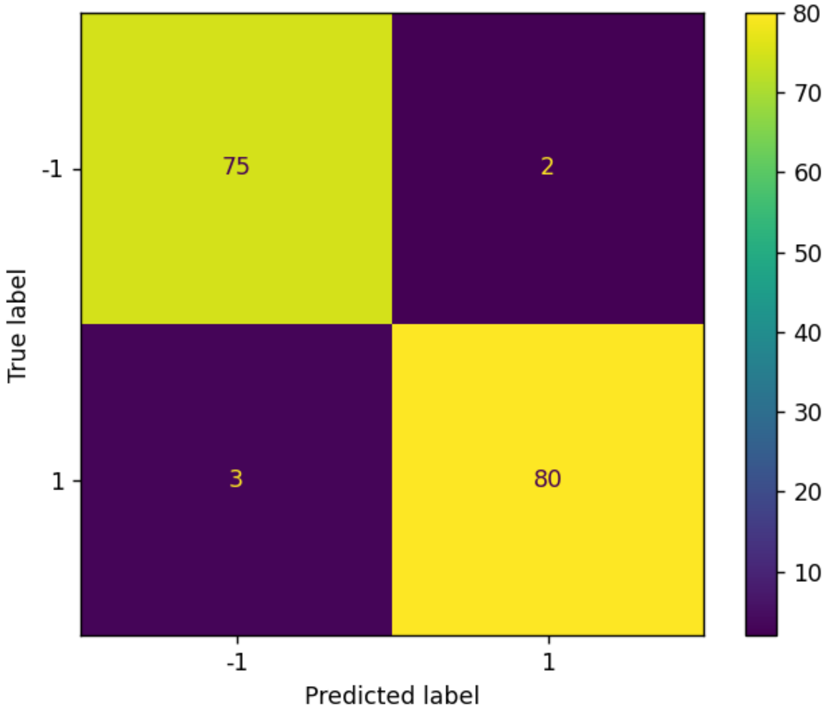
5.5 Оптимизируем подбор гиперпараметров через готовую библиотеку optuna, использующую алгоритм байесовской оптимизации.

5.6 Измерим качество на валидационной выборке.

Процесс подбора гиперпараметров продемонстрирован ниже:



```
[I 2025-05-26 22:30:34,348] Trial 436 finished with value: 0.7962962962962963 and parameters: {'C': 0.01648010660728377, 'gamma': 1.0670582819364565}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,412] Trial 437 finished with value: 0.7777777777777778 and parameters: {'C': 0.023269198861802138, 'gamma': 3.2088978144280054}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,481] Trial 438 finished with value: 0.7962962962962963 and parameters: {'C': 0.010623468372813692, 'gamma': 1.0236071687414274}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,549] Trial 439 finished with value: 0.7962962962962963 and parameters: {'C': 0.019976393518270474, 'gamma': 1.0018415741314064}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,624] Trial 440 finished with value: 0.8009259259259259 and parameters: {'C': 0.014882312856934973, 'gamma': 1.0001584205786946}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,696] Trial 441 finished with value: 0.7962962962962963 and parameters: {'C': 0.027193927144461297, 'gamma': 1.0460139381722002}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,767] Trial 442 finished with value: 0.7916666666666666 and parameters: {'C': 0.04330650098422181, 'gamma': 1.1332493087852589}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,836] Trial 443 finished with value: 0.7962962962962963 and parameters: {'C': 0.033862176493932014, 'gamma': 1.0277776297873702}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:34,905] Trial 444 finished with value: 0.7962962962962963 and parameters: {'C': 0.017585290789646917, 'gamma': 1.0654427753657991}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,025] Trial 445 finished with value: 0.7962962962962963 and parameters: {'C': 0.012954637623566611, 'gamma': 1.0274683779455804}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,108] Trial 446 finished with value: 0.7962962962962963 and parameters: {'C': 0.0666309211599394, 'gamma': 1.0928785763345228}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,188] Trial 447 finished with value: 0.8009259259259259 and parameters: {'C': 0.02263663629064064, 'gamma': 1.000913955219794}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,260] Trial 448 finished with value: 0.8009259259259259 and parameters: {'C': 0.027542271030819066, 'gamma': 1.0007963973777732}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,333] Trial 449 finished with value: 0.7962962962962963 and parameters: {'C': 0.009644496421085075, 'gamma': 1.0438199298689073}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,407] Trial 450 finished with value: 0.8009259259259259 and parameters: {'C': 0.01785443993026085, 'gamma': 1.0001289065094408}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,483] Trial 451 finished with value: 0.7962962962962963 and parameters: {'C': 0.0033107806352046486, 'gamma': 1.0469123125403943}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,555] Trial 452 finished with value: 0.7962962962962963 and parameters: {'C': 0.03701242351874582, 'gamma': 1.0770208243632289}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,631] Trial 453 finished with value: 0.7962962962962963 and parameters: {'C': 0.05136016653540497, 'gamma': 1.0266218379329497}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,699] Trial 454 finished with value: 0.7824074074074074 and parameters: {'C': 0.014182060438789262, 'gamma': 1.489844442531526}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,772] Trial 455 finished with value: 0.7962962962962963 and parameters: {'C': 0.004654155266214623, 'gamma': 1.0490682352166671}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,842] Trial 456 finished with value: 0.7962962962962963 and parameters: {'C': 0.023633165347805882, 'gamma': 1.025240214678088}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,907] Trial 457 finished with value: 0.8009259259259259 and parameters: {'C': 0.007672365859476256, 'gamma': 1.0012898438573357}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:35,986] Trial 458 finished with value: 0.7962962962962963 and parameters: {'C': 0.03077508922113288, 'gamma': 1.0637149724947754}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,101] Trial 459 finished with value: 0.7962962962962963 and parameters: {'C': 0.019521453344355857, 'gamma': 1.0283602439486832}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,170] Trial 460 finished with value: 0.7916666666666666 and parameters: {'C': 0.011366480726021257, 'gamma': 1.104874874864298}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,244] Trial 461 finished with value: 0.7962962962962963 and parameters: {'C': 0.016194833780775124, 'gamma': 1.0256034831733436}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,313] Trial 462 finished with value: 0.7962962962962963 and parameters: {'C': 0.025594570730173325, 'gamma': 1.0641391751226836}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,438] Trial 463 finished with value: 0.7962962962962963 and parameters: {'C': 0.09150243248660578, 'gamma': 1.001558758827716}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,526] Trial 464 finished with value: 0.8009259259259259 and parameters: {'C': 0.03942734692837103, 'gamma': 1.0005329595943653}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,604] Trial 465 finished with value: 0.7962962962962963 and parameters: {'C': 0.021609073697767392, 'gamma': 1.0461590055190129}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,681] Trial 466 finished with value: 0.7962962962962963 and parameters: {'C': 0.028505330766416963, 'gamma': 1.026817531619684}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,751] Trial 467 finished with value: 0.7962962962962963 and parameters: {'C': 0.012538199239155187, 'gamma': 1.078765274228441}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,822] Trial 468 finished with value: 0.7962962962962963 and parameters: {'C': 0.01802460412355338, 'gamma': 1.0244913451443196}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,895] Trial 469 finished with value: 0.7962962962962963 and parameters: {'C': 0.04739703696517935, 'gamma': 1.0501611996296487}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:36,966] Trial 470 finished with value: 0.7916666666666666 and parameters: {'C': 59.04759388392481, 'gamma': 1.0013140158041156}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,042] Trial 471 finished with value: 0.8009259259259259 and parameters: {'C': 0.03449944575881906, 'gamma': 1.0000530817474793}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,159] Trial 472 finished with value: 0.7962962962962963 and parameters: {'C': 0.015132313366721165, 'gamma': 1.0468908414116405}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,241] Trial 473 finished with value: 0.8009259259259259 and parameters: {'C': 0.010048407810917778, 'gamma': 1.0001265637855856}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,309] Trial 474 finished with value: 0.7916666666666666 and parameters: {'C': 0.02283333290980271, 'gamma': 1.1551141080048983}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,385] Trial 475 finished with value: 0.7962962962962963 and parameters: {'C': 0.030396789056852856, 'gamma': 1.028178388861408}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,450] Trial 476 finished with value: 0.7777777777777778 and parameters: {'C': 0.019263544309512645, 'gamma': 2.505291272556596}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,521] Trial 477 finished with value: 0.7962962962962963 and parameters: {'C': 0.0063897281079606325, 'gamma': 1.0765717161798025}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,599] Trial 478 finished with value: 0.7962962962962963 and parameters: {'C': 0.06968641723741338, 'gamma': 1.02610971350661 3}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,670] Trial 479 finished with value: 0.7962962962962963 and parameters: {'C': 0.024843065960629157, 'gamma': 1.0499149041514892}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,739] Trial 480 finished with value: 0.7962962962962963 and parameters: {'C': 0.014488968782767363, 'gamma': 1.099153943591109}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,807] Trial 481 finished with value: 0.7777777777777778 and parameters: {'C': 0.03862394463950872, 'gamma': 3.8779024198373735}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,877] Trial 482 finished with value: 0.7962962962962963 and parameters: {'C': 0.011611873563737177, 'gamma': 1.0301109006713522}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:37,954] Trial 483 finished with value: 0.8009259259259259 and parameters: {'C': 0.019389318277514857, 'gamma': 1.000715887776055}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:38,025] Trial 484 finished with value: 0.8009259259259259 and parameters: {'C': 0.05593115961955196, 'gamma': 1.0000170985596197}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:38,102] Trial 485 finished with value: 0.7962962962962963 and parameters: {'C': 0.031605298400374196, 'gamma': 1.0611689883406055}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:38,206] Trial 486 finished with value: 0.7962962962962963 and parameters: {'C': 0.18280257927447696, 'gamma': 1.0288930703942338}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:38,294] Trial 487 finished with value: 0.7962962962962963 and parameters: {'C': 0.008978082271934078, 'gamma': 1.0464392551874535}. Best is trial 51 with value: 0.8009259259259259.
[I 2025-05-26 22:30:38,379] Trial 488 finished with value: 0.7962962962962963 and parameters: {'C': 0.0021668366815725186, 'gamma': 1.0256714804949465}. Best is trial 51 with value: 0.8009259259259259.
```



В итоге мы получили модель с точность (accurarcy) равным 0.78 и :

- Лучшим C = 0.02910936784038153,
- Лучшей гаммой = 1.0006548806873106,
- Лучшей точностью = 0.8009259259259259

Как можно заметить лучшей гаммой было признано значение равное 1. Это указывает на то, что данные хорошо разделены по признакам, но при этом требуется некоторая гибкость при подсчёте. Параметр же регуляризации довольно мал - 0.02, что говорит о том, что данные содержат шум.