|  |  |  |
| --- | --- | --- |
|  | Изображение выглядит как зарисовка, рисунок, символ, эмблема  Автоматически созданное описание |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |
| **Институт комплексной безопасности и специального приборостроения**  Кафедра КБ-14 «Технологии искусственного интеллекта и анализ данных»  Дисциплина «Введение в машинное обучение» | | |

**Отчёт**

«Практическая работа №3»

Выполнил:

Студент 3 курса

Группа БСБО-06-22

Шифр 22Б1240

Белявцева Екатерина Александровна

Москва, 2024

# Задание

Разработать SVM-классификатор (метод опорных векторов) для набора данных. В качестве языка программирования использовать Pyhton. Разбить выборку на обучающую и тестовую. Обучить, проверить качество классификатора на обучающей и тестовой выборках: рассчитать общую точность, Recall, Precision, F1-measure. Оценить число опорных векторов. Перебор по сетке (grid search).

Рассмотреть различные типы ядра (линейное, RBF, полиномиальное, сигмоидное (тангенсальное)), различные сочетания значений параметра регуляризации С и параметров ядра.

# Ход работы

Для выполнения первой части был использован язык программирования Python в приложении PyCharm. Для корректной работы кода был импортирован текстовый документ «Ирисы». В коде выборка была разбита на обучающую и тестовую, была обучена и проверено качество классификатора (были рассчитаны общая точность, Recall, Precision и F1-measure). Также было оценено число опорных векторов. После этого код был немного изменён и добавлен перебор по сетке (grid search). Листинг кода на данном этапе представлен ниже (см. Листинг 1).

Листинг 1. Код на первом этапе

|  |
| --- |
| import pandas as pd from sklearn.model\_selection import train\_test\_split, GridSearchCV from sklearn.svm import SVC from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score, classification\_report  # Загружаем данные из текстового файла data = pd.read\_csv('Ирисы.txt', delimiter='\t', encoding='windows-1251')  # Заменяем запятые на точки в числовых данных data = data.replace(',', '.', regex=True) # Разделение данных на признаки (X) и метки (y) X = data.iloc[:, :-1] # Все столбцы, кроме последнего — это признаки y = data.iloc[:, -1] # Последний столбец — метки классов  # Преобразуем все числовые данные к float X = X.astype(float)  # Разделяем данные на обучающую и тестовую выборки X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)  # Параметры для перебора по сетке param\_grid = {  'C': [0.1, 1, 10], # Параметр регуляризации  'kernel': ['linear'], # Используем только линейное ядро }  # Создаем и обучаем модель SVM с заданными параметрами svm\_classifier = SVC(kernel='linear')  # Создаем объект GridSearchCV grid\_search = GridSearchCV(estimator=svm\_classifier, param\_grid=param\_grid,  scoring='accuracy', cv=5, n\_jobs=-1)  # Обучаем модель с Grid Search grid\_search.fit(X\_train, y\_train)  # Получаем наилучшие параметры и результаты best\_params = grid\_search.best\_params\_ best\_model = grid\_search.best\_estimator\_  # Делаем предсказания на тестовой выборке y\_pred = best\_model.predict(X\_test)  # Рассчитываем метрики для проверки качества модели с zero\_division accuracy = accuracy\_score(y\_test, y\_pred) # Общая точность precision = precision\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя точность (Precision) recall = recall\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя полнота (Recall) f1 = f1\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя F1-метрика  # Выводим результаты print(f"Общая точность модели: {accuracy:.2f}") print(f"Средняя точность (Precision): {precision:.2f}") print(f"Средняя полнота (Recall): {recall:.2f}") print(f"Средняя F1-метрика: {f1:.2f}") print("Лучшие параметры:", best\_params)  # Вывод отчета по классификации print("Отчет по классификации:\n", classification\_report(y\_test, y\_pred)) |

На этом этапе вывод выглядит следующим образом (см. рис. 1).

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 1. Вывод кода на первом этапе

Далее были проведены изменения, чтобы рассмотреть различные типы ядер, различные сочетания значений параметра регуляризации С и параметров ядра (см. Листинг 2).

Листинг 2. Окончательный код

|  |
| --- |
| import pandas as pd from sklearn.model\_selection import train\_test\_split, GridSearchCV from sklearn.svm import SVC from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score, classification\_report  # Загружаем данные из текстового файла data = pd.read\_csv('Ирисы.txt', delimiter='\t', encoding='windows-1251')  # Заменяем запятые на точки в числовых данных data = data.replace(',', '.', regex=True)  # Разделение данных на признаки (X) и метки (y) X = data.iloc[:, :-1] # Все столбцы, кроме последнего — это признаки y = data.iloc[:, -1] # Последний столбец — метки классов  # Преобразуем все числовые данные к float X = X.astype(float)  # Разделяем данные на обучающую и тестовую выборки X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)   # Функция для оценки различных моделей SVM def evaluate\_svm(kernel, C=1.0, degree=3, gamma='scale'):  # Определяем параметры для перебора по сетке  param\_grid = {  'C': [0.1, 1, 10], # Параметр регуляризации  'kernel': ['linear', 'rbf', 'poly', 'sigmoid'], # Различные ядра  'degree': [2, 3, 4], # Степень для полиномиального ядра  'gamma': ['scale', 'auto'] # Параметр для RBF и сигмоидного ядра  }   # Создаем и обучаем модель SVM с заданными параметрами  svm\_classifier = SVC(kernel=kernel, C=C, degree=degree, gamma=gamma)  # Создаем объект GridSearchCV  grid\_search = GridSearchCV(estimator=svm\_classifier, param\_grid=param\_grid,  scoring='accuracy', cv=5, n\_jobs=-1)   # Обучаем модель с Grid Search  grid\_search.fit(X\_train, y\_train)   # Получаем наилучшие параметры и результаты  best\_params = grid\_search.best\_params\_  best\_model = grid\_search.best\_estimator\_   # Делаем предсказания на тестовой выборке  y\_pred = best\_model.predict(X\_test)   # Рассчитываем метрики для проверки качества модели с zero\_division  accuracy = accuracy\_score(y\_test, y\_pred) # Общая точность  precision = precision\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя точность (Precision)  recall = recall\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя полнота (Recall)  f1 = f1\_score(y\_test, y\_pred, average='weighted', zero\_division=0) # Средняя F1-метрика   # Оценка количества опорных векторов  n\_support\_vectors = best\_model.n\_support\_  total\_support\_vectors = sum(n\_support\_vectors)   # Выводим результаты  print(f"Ядро: {kernel}, C: {C}, degree: {degree}, gamma: {gamma}")  print(f"Общая точность модели: {accuracy:.2f}")  print(f"Средняя точность (Precision): {precision:.2f}")  print(f"Средняя полнота (Recall): {recall:.2f}")  print(f"Средняя F1-мерa: {f1:.2f}")  print(f"Число опорных векторов: {total\_support\_vectors}\n")  print("Лучшие параметры:", best\_params)  print(f"Отчет по классификации для ядра {kernel}:\n", classification\_report(y\_test, y\_pred))  print("=" \* 50)    # Сценарии для различных ядер и параметров kernels = ['linear', 'rbf', 'poly', 'sigmoid'] C\_values = [0.1, 1, 10] degrees = [2, 3, 4] # Для полиномиального ядра gamma\_values = ['scale', 'auto'] # Для RBF и сигмоидного ядра  # Линейное ядро for C in C\_values:  evaluate\_svm(kernel='linear', C=C)  # RBF ядро for C in C\_values:  for gamma in gamma\_values:  evaluate\_svm(kernel='rbf', C=C, gamma=gamma)  # Полиномиальное ядро for C in C\_values:  for degree in degrees:  evaluate\_svm(kernel='poly', C=C, degree=degree)  # Сигмоидное ядро for C in C\_values:  for gamma in gamma\_values:  evaluate\_svm(kernel='sigmoid', C=C, gamma=gamma) |

Для проверки правильности количества выведенных значений был проведён следующий анализ:

1. для каждого значения C для линейного ядра выполняется вызов функции evaluate\_svm. C принимает 3 значения (0.1; 1; 10). Следовательно, будет 3 комбинации;
2. для каждого значения C для RBF ядра и для каждого значения gamma выполняется вызов функции evaluate\_svm. С также принимает 3 значения (0.1; 1; 10), а gamma принимает ещё 2 значения (scale, auto). Итого для RBF ядра будет 6 значений;
3. для каждого значения C для полиноминального ядра и для каждого значения degree выполняется вызов функции evaluate\_svm. C всё также принимает 3 значения (0.1; 1; 10), а degree – 3 значения (2; 3; 4). В сумме выходит 9 комбинаций;
4. для сигмоидного ядра точно также, как и для RBF ядра, С – 3 значения (0.1; 1; 10), gamma – 2 значения (scale, auto). Итого – также 6 значений.

Итого общее количество комбинаций будет составлять 24 штуки. Далее будет приведён вывод в виде Листинга 4. Помимо этого вышеописанный код выводит лучшие параметры и отчёт о классификации для каждой комбинации параметров.

Листинг 4. Вывод комбинаций параметров для ядер

|  |
| --- |
| Ядро: linear, C: 0.1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра linear:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: linear, C: 1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра linear:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: linear, C: 10, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра linear:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 0.1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 0.1, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 1, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 10, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: rbf, C: 10, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра rbf:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 0.1, degree: 2, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 0.1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 0.1, degree: 4, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 1, degree: 2, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 1, degree: 4, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 10, degree: 2, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 10, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: poly, C: 10, degree: 4, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра poly:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 0.1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 0.1, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 1, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 1, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 10, degree: 3, gamma: scale  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ==================================================  Ядро: sigmoid, C: 10, degree: 3, gamma: auto  Общая точность модели: 0.98  Средняя точность (Precision): 0.98  Средняя полнота (Recall): 0.98  Средняя F1-мерa: 0.98  Число опорных векторов: 19  Лучшие параметры: {'C': 1, 'degree': 3, 'gamma': 'scale', 'kernel': 'poly'}  Отчет по классификации для ядра sigmoid:  precision recall f1-score support  Iris-setosa 1.00 1.00 1.00 19  Iris-versicolor 1.00 0.92 0.96 13  Iris-virginica 0.93 1.00 0.96 13  accuracy 0.98 45  macro avg 0.98 0.97 0.97 45  weighted avg 0.98 0.98 0.98 45  ================================================== |