

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский Государственный технический университет»  
Кафедра ИИТ

Лабораторная работа №3  
По дисциплине «Основы машинного обучения»  
Тема: **«Сравнение классических методов классификации»**

Выполнила:  
Студентка 3 курса  
Группы АС-65  
Сунцова М. Д.  
Проверил:  
Крощенко А. А.

**Цель:** на практике сравнить работу нескольких алгоритмов классификации, таких как метод k-ближайших соседей (k-NN), деревья решений и метод опорных векторов (SVM). Научиться подбирать гиперпараметры моделей и оценивать их влияние на результат.

## Вариант 8

- Seeds
- Классифицировать семена на три сорта пшеницы (Kama, Rosa, Canadian) на основе их геометрических параметров

- **Задания:**

1. Загрузите и стандартизируйте данные;
2. Разделите выборку на обучающую и тестовую;

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.decomposition import PCA
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

df = pd.read_csv("seeds_dataset.txt", delim_whitespace=True, header=None)
print(df.head(5))

X = df.iloc[:, :-1]
y = df.iloc[:, -1].astype(int)

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.3, random_state=42
)
```

	0	1	2	3	4	5	6	7
0	15.26	14.84	0.8710	5.763	3.312	2.221	5.220	1
1	14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1
2	14.29	14.09	0.9050	5.291	3.337	2.699	4.825	1
3	13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1
4	16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1

3. Обучите три классификатора;
4. Сравните общую точность (accuracy) всех трех моделей;

```
models = {
    "k-NN": KNeighborsClassifier(n_neighbors=5),
    "Decision Tree": DecisionTreeClassifier(max_depth=5, random_state=42),
    "SVM": SVC(kernel='rbf', C=1, gamma='scale', random_state=42)
}

accuracies = {}
for name, model in models.items():
```

```

model.fit(X_train, y_train)
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
accuracies[name] = acc
print(f"{name} Accuracy: {acc:.4f}")

```

```

best_model_name = max(accuracies, key=accuracies.get)
best_model = models[best_model_name]
print(f"\nЛучшая модель: {best_model_name} с точностью {accuracies[best_model_name]:.4f}")

```

```

k-NN Accuracy: 0.9048
Decision Tree Accuracy: 0.8571
SVM Accuracy: 0.9206

Лучшая модель: SVM с точностью 0.9206

```

5. Визуализируйте данные в 2D (например, с помощью PCA), раскрасив точки в соответствии с предсказаниями лучшей модели.

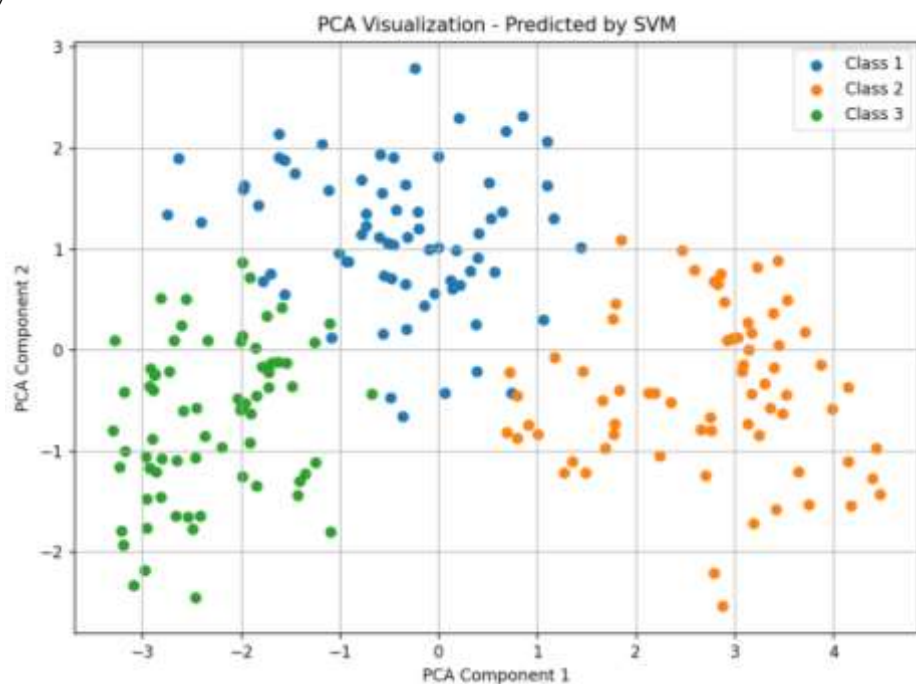
```

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
y_pred_best = best_model.predict(X_scaled)

plt.figure(figsize=(8, 6))
for label in np.unique(y_pred_best):
    plt.scatter(
        X_pca[y_pred_best == label, 0],
        X_pca[y_pred_best == label, 1],
        label=f"Class {label}"
    )

plt.title(f"PCA Visualization - Predicted by {best_model_name}")
plt.xlabel("PCA Component 1")
plt.ylabel("PCA Component 2")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```



**Вывод:** график показывает, что модель SVM хорошо разделяет три класса: точки разных цветов образуют отдельные группы, почти не перекрывающиеся. Это значит, что классификация прошла успешно.