# Лабораторная работа 2

## Анализ DDoS атак моделями машинного обучения

Дан датасет DDoS атак

https://www.kaggle.com/code/chitrakshsingh/ddos-sdn

Разработать и протестировать модели машинного обучения для классификации сетевого трафика и выявления DDoS-атак на основе датасета DDoS-SDN. Провести сравнительный анализ различных алгоритмов.

### Этапы выполнения

### 1. Подготовка данных

1. **Загрузка данных**
   - Скачать датасет с Kaggle и загрузить в среду разработки (например, Jupyter Notebook или PyCharm).
   - Использовать библиотеки pandas и numpy для обработки данных.
2. **Анализ данных**
   - Провести анализ структуры данных (количество строк, столбцов, наличие пропущенных значений).
   - Определить целевую переменную и признаки.
3. **Предобработка данных**
   - Обработать пропущенные значения (если есть).
   - Закодировать категориальные переменные (если они присутствуют).
   - Нормализовать числовые признаки (если необходимо).
   - Разделить данные на обучающую и тестовую выборки (train_test_split).

### 2. Обучение моделей

Реализовать и обучить следующие модели:

1. **Наивный Байесовский классификатор (Naïve Bayes)**
   - Использовать GaussianNB или MultinomialNB из sklearn.naive_bayes.
2. **Логистическая регрессия (Logistic Regression)**
   - Использовать LogisticRegression из sklearn.linear_model.
3. **Метод опорных векторов (Support Vector Machine, SVM)**
   - Использовать SVC из sklearn.svm с разными ядрами (linear, rbf).
4. **Метод k-ближайших соседей (k-Nearest Neighbors, k-NN)**
   - Использовать KNeighborsClassifier из sklearn.neighbors.
   - Подобрать оптимальное k с помощью кросс-валидации.
5. **Дерево решений (Decision Tree)**
   - Использовать DecisionTreeClassifier из sklearn.tree.
   - Провести настройку гиперпараметров (max_depth, criterion).
6. **Случайный лес (Random Forest)**
   - Использовать RandomForestClassifier из sklearn.ensemble.
   - Настроить параметры (n_estimators, max_depth).
7. **Градиентный бустинг (XGBoost)**

- Использовать XGBClassifier из xgboost.
- Настроить параметры (learning_rate, n_estimators, max_depth).

8. **CatBoost**
   - Использовать CatBoostClassifier из catboost.
   - Подобрать параметры (iterations, depth, learning_rate).
9. **AdaBoost**
   - Использовать AdaBoostClassifier из sklearn.ensemble.
   - Настроить параметры (n_estimators, learning_rate).

**3. Оценка моделей**

1. **Метрики качества**
   - Оценить модели по метрикам:
     - accuracy
     - precision
     - recall
     - F1-score
     - ROC-AUC (для вероятностных моделей)
2. **Кросс-валидация**
   - Провести кросс-валидацию (например, StratifiedKFold) для проверки устойчивости моделей.
3. **Матрица ошибок (Confusion Matrix)**
   - Построить матрицу ошибок (confusion_matrix) для анализа ошибок классификации.

**4**. **Визуализация результатов**

1. Построить графики:
   - Сравнение точности (accuracy) моделей.
   - График ROC-кривых для нескольких моделей.
   - Матрицы ошибок для лучших моделей.

**5. Анализ и выводы**

1. Сравнить результаты всех моделей, указав:
   - Какая модель показала наилучшую точность?
   - Какой алгоритм быстрее обучается?
   - Какие модели лучше справляются с данной задачей?
   - Какие признаки наиболее значимы?
2. Написать выводы о применимости различных алгоритмов для классификации DDoS-атак.