



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)

Кафедра прикладной математики

ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 7
по дисциплине «Технологии и инструментальный анализ
больших данных»

Выполнил студент группы ИКБО-20-21
Проверил ассистент кафедры ПМ ИИТ

Сидоров С.Д.
Тетерин Н.Н.

Москва 2024

Практическая работа

1. Найти данные для задачи классификации или для задачи регрессии (данные не должны повторяться в группе).

Листинг 1:

```
# Загрузка датасета breast cancer из sklearn
cancer_data = datasets.load_breast_cancer()

# Преобразование в DataFrame
data = pd.DataFrame(data=cancer_data.data,
                    columns=cancer_data.feature_names)
data['target'] = cancer_data.target

# Проверка структуры данных
print(data.head())

# Выделение признаков и целевой переменной
X = data.drop(columns=['target'])
y = data['target']

# Разделение на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.9, random_state=42)
```

2. Реализовать баггинг.

Листинг 2:

```
# Инициализация модели баггинга
bagging_clf = BaggingClassifier(estimator=DecisionTreeClassifier(),
                               n_estimators=50, random_state=42)

# Засекаем время обучения
start_time = time.time()

# Обучение модели
bagging_clf.fit(X_train, y_train)

# Время работы модели
bagging_time = time.time() - start_time

# Предсказания на тестовой выборке
y_pred_bagging = bagging_clf.predict(X_test)

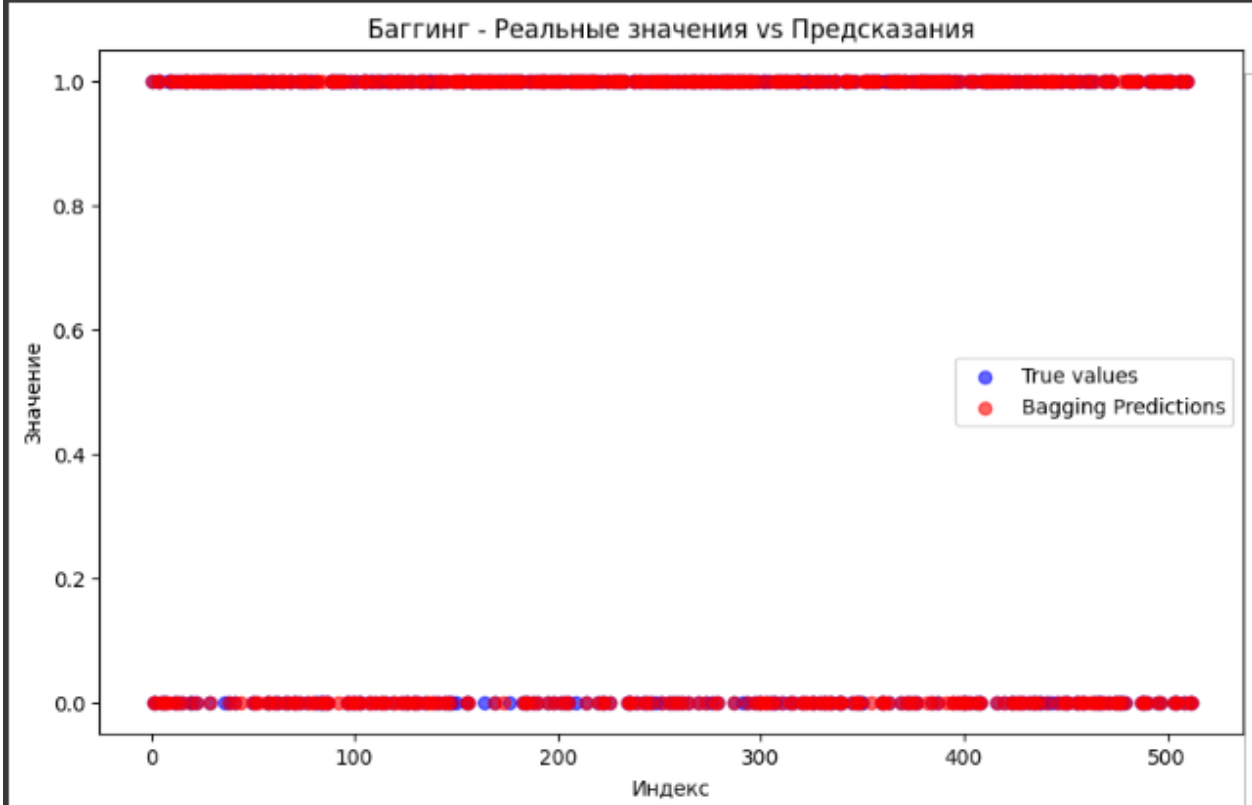
# Расчет точности (accuracy) и коэффициента детерминации ( $R^2$ )
bagging_accuracy = accuracy_score(y_test, y_pred_bagging)
bagging_r2 = r2_score(y_test, y_pred_bagging)

# Вывод результатов
print(f"Баггинг - Время работы: {bagging_time:.4f} секунд")
print(f"Баггинг - Точность: {bagging_accuracy:.4f}")
print(f"Баггинг - Коэффициент детерминации  $R^2$ : {bagging_r2:.4f}")

# Построение графика для баггинга
```

```
plt.figure(figsize=(10, 6))
plt.scatter(range(len(y_test)), y_test, color='blue', label='True
values', alpha=0.6)
plt.scatter(range(len(y_pred_bagging)), y_pred_bagging, color='red',
label='Bagging Predictions', alpha=0.6)
plt.title('Баггинг - Реальные значения vs Предсказания')
plt.xlabel('Индекс')
plt.ylabel('Значение')
plt.legend()
plt.show()
```

Баггинг - Время работы: 0.4488 секунд
Баггинг - Точность: 0.9201
Баггинг - Коэффициент детерминации R^2 : 0.6594



3. Реализовать бустинг на тех же данных, что использовались для баггинга.

Листинг 3:

```
# Инициализация модели градиентного бустинга
boosting_clf = GradientBoostingClassifier(n_estimators=50,
random_state=42)

# Засекаем время обучения
start_time = time.time()

# Обучение модели
boosting_clf.fit(X_train, y_train)

# Время работы модели
boosting_time = time.time() - start_time
```

```
# Предсказания на тестовой выборке
y_pred_boosting = boosting_clf.predict(X_test)

# Расчет точности (accuracy) и коэффициента детерминации ( $R^2$ )
boosting_accuracy = accuracy_score(y_test, y_pred_boosting)
boosting_r2 = r2_score(y_test, y_pred_boosting)
```

```
Бустинг - Время работы: 0.2500 секунд
Бустинг - Точность: 0.9045
Бустинг - Коэффициент детерминации  $R^2$ : 0.5930
```

4. Сравнить результаты работы алгоритмов (время работы и качество моделей). Сделать выводы.

Листинг 4:

```
# Сравнение времени работы и качества моделей
print(f"Сравнение времени работы:")
print(f"Баггинг: {bagging_time:.4f} секунд")
print(f"Бустинг: {boosting_time:.4f} секунд")

print("\nСравнение точности:")
print(f"Баггинг - Точность: {bagging_accuracy:.4f}")
print(f"Бустинг - Точность: {boosting_accuracy:.4f}")

print("\nСравнение коэффициента детерминации  $R^2$ :")
print(f"Баггинг -  $R^2$ : {bagging_r2:.4f}")
print(f"Бустинг -  $R^2$ : {boosting_r2:.4f}")
```

```
Сравнение времени работы:
Баггинг: 0.4488 секунд
Бустинг: 0.2500 секунд

Сравнение точности:
Баггинг - Точность: 0.9201
Бустинг - Точность: 0.9045

Сравнение коэффициента детерминации  $R^2$ :
Баггинг -  $R^2$ : 0.6594
Бустинг -  $R^2$ : 0.5930
```

Выводы:

Время работы: Баггинг оказался медленнее

Качество предсказаний: на наборе данных **BreastCancer** обе модели показали одинаково высокие результаты. Однако бустинг на более сложных данных может быть точнее, так как его механизмы направлены на улучшение ошибок, тогда как баггинг создает независимые модели, не исправляя ошибки друг друга.