Здобняков Фёдор Андреевич ИУ5-64Б

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder, StandardScaler
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.impute import SimpleImputer
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, r2_score

data = pd.read_csv('salaries.csv')

print("Пропуски в данных:")
print(data.isnull().sum())

print("\nНазвания столбцов в данных:")
print(data.columns)

# Заполнение пропусков
imputer = SimpleImputer(strategy='mean')
data['salary_in_usd'] = imputer.fit_transform(data[['salary_in_usd']])

# Размерности
X = data.drop(columns=['salary_in_usd'])
y = data['salary_in_usd']  # Целевая переменная

print("\nРазмеры X и y:")
print(X.shape, y.shape)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```python
# Определение числовых и категориальных столбцов
numeric_features = ['work_year', 'salary', 'remote_ratio']
categorical_features = ['experience_level', 'employment_type', 'job_title',
                        'salary_currency', 'employee_residence',
                        'company_location', 'company_size']

# Преобразование числовых и категориальных признаков
numeric_transformer = Pipeline(steps=[
    ('imputer', SimpleImputer(strategy='mean')),
    ('scaler', StandardScaler())
])

categorical_transformer = Pipeline(steps=[
    ('imputer', SimpleImputer(strategy='constant', fill_value='missing')),
    ('onehot', OneHotEncoder(handle_unknown='ignore'))
])

preprocessor = ColumnTransformer(
    transformers=[
        ('num', numeric_transformer, numeric_features),
        ('cat', categorical_transformer, categorical_features)
    ])

# Линейная регрессия
lr_model = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', LinearRegression())
])

# Градиентный бустинг
gb_model = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', GradientBoostingRegressor())
])
```

```python
print("Обучаем линейную регрессию...")
lr_model.fit(X_train, y_train)
print("Обучение завершено.")

print("Обучаем градиентный бустинг...")
gb_model.fit(X_train, y_train)
print("Обучение завершено.")

# Прогнозы
y_pred_lr = lr_model.predict(X_test)
y_pred_gb = gb_model.predict(X_test)

# Оценка качества моделей
mse_lr = mean_squared_error(y_test, y_pred_lr)
r2_lr = r2_score(y_test, y_pred_lr)

mse_gb = mean_squared_error(y_test, y_pred_gb)
r2_gb = r2_score(y_test, y_pred_gb)

# Вывод метрик
print(f"Linear Regression MSE: {mse_lr}, R2: {r2_lr}")
print(f"Gradient Boosting MSE: {mse_gb}, R2: {r2_gb}")
```

```
Пропуски в данных:
work_year             0
experience_level      0
employment_type       0
job_title             0
salary                0
salary_currency       0
salary_in_usd         0
employee_residence    0
remote_ratio          0
company_location      0
company_size          0
dtype: int64

Названия столбцов в данных:
Index(['work_year', 'experience_level', 'employment_type', 'job_title',
       'salary', 'salary_currency', 'salary_in_usd', 'employee_residence',
       'remote_ratio', 'company_location', 'company_size'],
      dtype='object')

Размеры X и y:
(88584, 10) (88584,)
Обучаем линейную регрессию...
Обучение завершено.
Обучаем градиентный бустинг...
Обучение завершено.
Linear Regression MSE: 8027132101.067358, R2: -0.45337942798027764
Gradient Boosting MSE: 10131554.336249944, R2: 0.9981655973191457
```

MSE (Среднеквадратичная ошибка)
Что это?
MSE измеряет среднеквадратичное отклонение предсказанных значений от реальных. Это показывает, насколько сильно предсказания модели отклоняются от фактических значений.

Почему использовал?

MSE является важной метрикой для регрессионных задач, так как она показывает, насколько ошибка модели велика. Чем меньше значение MSE, тем лучше модель предсказывает.

Для линейной регрессии MSE составил 8,027,132,101.07, что является очень высоким значением и указывает на то, что модель плохо справляется с предсказаниями.

Для градиентного бустинга MSE составил 10,131,554.34, что значительно ниже, и модель показывает гораздо лучшие результаты.

R^2 (Коэффициент детерминации)
Что это?
R^2 — это метрика, показывающая, какая доля вариации зависимой переменной объясняется моделью. Он измеряет, насколько хорошо модель предсказывает результат по сравнению с базовой моделью, которая предсказывает только среднее значение.

Почему использовал?
R^2 позволяет понять, насколько хорошо модель объясняет данные. Высокое значение R^2 свидетельствует о том, что модель эффективно улавливает закономерности в данных.

Результат:
Для линейной регрессии R^2 составил -0.453, что означает, что модель объясняет менее половины вариации в данных и не лучше, чем простое среднее значение. Это указывает на плохую модель.
Для градиентного бустинга R^2 составил 0.998, что практически идеально, и говорит о том, что модель отлично объясняет данные.

Линейная регрессия:
Результаты линейной регрессии с высоким MSE и отрицательным R^2 говорят о том, что модель плохо справляется с задачей. Это может быть связано с тем, что данные имеют сложные, нелинейные зависимости, которые линейная регрессия не может уловить.

Важно отметить, что линейная регрессия эффективна только в случае, если существует линейная зависимость между признаками и целевой переменной. В данном случае модель не смогла бы объяснить вариацию данных, что подтверждается отрицательным значением R^2.

Градиентный бустинг:
Результаты градиентного бустинга с низким MSE и высоким R^2 показывают, что эта модель значительно лучше справляется с задачей. Она эффективно объясняет данные и может находить сложные закономерности, что делает ее более подходящей для таких задач, чем линейная регрессия.
Показатель R^2 близкий к 1 указывает на то, что градиентный бустинг практически идеально объясняет вариацию в данных, что делает модель отличным выбором для этой задачи.