

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра системного проектування

Звіт

Про виконання лабораторної роботи №1
3 курсу «Системи машинного навчання»
Вступ в машинне навчання та Scikit-learn

Виконав:
Студент групи ФеС-32
Бойко Кирило

Перевірив:
Доцент Колич І.І.

Львів 2024

Мета: Ознайомитися з базовими поняттями машинного навчання та бібліотекою Scikitlearn.

Хід роботи:

1. Завантаження готових наборів даних з Scikit-learn (також завантажуються та імпортуються усі необхідні бібліотеки для виконання лабораторної):

```
[1] import pandas as pd
    from sklearn.datasets import fetch_california_housing
    from sklearn.model_selection import train_test_split
    import numpy as np
    import random
    from sklearn.linear_model import LinearRegression
    from sklearn.metrics import r2_score
```

```
[2] housing_data = fetch_california_housing()
```

```
[3] df = pd.DataFrame(housing_data.data, columns=housing_data.feature_names)
    df['MedHouseVal'] = housing_data.target
```

```
[4] df.head(7)
```

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude	MedHouseVal
0	8.3252	41.0	6.984127	1.023810	322.0	2.555556	37.88	-122.23	4.526
1	8.3014	21.0	6.238137	0.971880	2401.0	2.109842	37.86	-122.22	3.585
2	7.2574	52.0	8.288136	1.073446	496.0	2.802260	37.85	-122.24	3.521
3	5.6431	52.0	5.817352	1.073059	558.0	2.547945	37.85	-122.25	3.413
4	3.8462	52.0	6.281853	1.081081	565.0	2.181467	37.85	-122.25	3.422
5	4.0368	52.0	4.761658	1.103627	413.0	2.139896	37.85	-122.25	2.697
6	3.6591	52.0	4.931907	0.951362	1094.0	2.128405	37.84	-122.25	2.992

2. Поділ даних на тренувальну та тестову вибірки:

```
[5] X = df.drop('MedHouseVal', axis=1)
    y = df['MedHouseVal']
```

```
[6] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

3. Написання функції для формули множинної лінійної регресії:

```
[7] def multiple_linear_regression(X, y, coefficients):
    return np.dot(X, coefficients)
```

4. Випадковий підбір коефіцієнтів:

```
[8] my_str = "Boiko"
    res = ''.join(format(ord(i), '08b') for i in my_str)
    my_seed = int(res) % 12345

[9] np.random.seed(my_seed)

[10] def mean_squared_error(y_true, y_pred):
      return np.mean((y_true - y_pred)**2)

[11] best_mse = float('inf')
      best_coefs = None
      for _ in range(1000):
          coefs = np.random.rand(X_train.shape[1])
          y_pred = multiple_linear_regression(X_train, y_train, coefs)
          mse = mean_squared_error(y_train, y_pred)
          if mse < best_mse:
              best_mse = mse
              best_coefs = coefs

[12] y_pred_test = multiple_linear_regression(X_test, y_test, best_coefs)
      r2 = r2_score(y_test, y_pred_test)
      print(f"r2: {r2}; mse: {mse}")

r2: -95.87355584942743; mse: 419128.77234035847
```

5. Навчання та оцінка простої моделі (наприклад, лінійна регресія):

```
[13] model = LinearRegression()
      model.fit(X_train, y_train)

LinearRegression()

[14] y_pred = model.predict(X_test)

[15] mse = mean_squared_error(y_test, y_pred)
      r2 = r2_score(y_test, y_pred)

      print(f"r2: {r2}; mse: {mse}")

r2: 0.595770232606166; mse: 0.5305677824766758
```

Висновок: Виконавши дану лабораторну роботу я отримав міцну основу для подальшого вивчення машинного навчання. А також, практичні навички, які дозволяють вирішувати більш складні задачі в майбутньому.