

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 №3
З курсу «Системи машинного навчання»
Регуляризація

Виконала:
Студентка групи ФЕС-32
Філь Дарина

Перевірив:
Доцент Колич І.І.

Львів 2024

Мета: Засвоїти основи регресійного аналізу з використанням регуляризації.

Інструменти: Python, Skikit-learn, Matplotlib, Seaborn.

Теоретичні відомості

Лінійна регресія

Лінійна регресія є основним методом машинного навчання для моделювання взаємозв'язків між змінними. Вона дозволяє передбачати значення залежної змінної на основі незалежних змінних.

Формула лінійної регресії

Формула лінійної регресії (з більш ніж однією незалежною змінною) виглядає так:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n$$

де:

- y — залежна змінна;
- β_0 — вільний член (intercept);
- β_1, \dots, β_n — коефіцієнти регресії для незалежної змінної;
- x_1, \dots, x_n — незалежні змінні;

Поліноміальна регресія

Поліноміальна регресія є узагальненням лінійної регресії, яка дозволяє моделювати нелінійні взаємозв'язки між змінними.

Формула поліноміальної регресії:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_d x^d$$

де:

- y — залежна змінна;
- β_0 — вільний член (intercept);
- β_1, \dots, β_n — коефіцієнти регресії для незалежної змінної;
- x_1, \dots, x_n — незалежні змінні;
- d — ступінь полінома;

Регуляризація

Регуляризація є технікою, яка використовується для запобігання перенавчанню (overfitting) моделей шляхом додавання штрафу до функції втрат. Основні методи регуляризації — це Ridge і Lasso.

Причини використання регуляризації:

1. **Запобігання перенавчанню:** регуляризація допомагає моделі загальнювати краще на нових даних, запобігаючи перенавчанню на тренувальних даних.
2. **Стабільність моделі:** регуляризація зменшує варіацію у прогнозах моделі, роблячи її більш стабільною.
3. **Інтерпретованість:** lasso регресія може використовуватися для відсічення неважливих змінних, що покращує інтерпретованість моделі.

Основні методи регуляризації

1. Ridge регресія (L2 регуляризація)

- **Визначення:** Ridge регресія додає штраф за величину коефіцієнтів регресії до функції втрат. Це допомагає зменшити величину коефіцієнтів, що може знизити ризик перенавчання.
- **Формула:** $Loss = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^m (\beta_j)^2$
де:
 - α – параметр регуляризації, який контролює величину штрафу;
 - β_j – коефіцієнт регресії;
- **Механізм:** Ridge регресія додає штраф у вигляді суми квадратів коефіцієнтів регресії. Це означає, що модель намагається зменшити величину коефіцієнтів, щоб уникнути перенавчання.
- **Переваги:**
 - Допомогає запобігти перенавчанню;
 - Підходить для моделей з великим числом ознак;
- **Недоліки:**
 - Не може виконувати відсічення ознак (значення коефіцієнтів регресії зменшуються, але не стають нульовими);

2. Lasso регресія (L1 регуляризація)

- **Визначення:** Lasso регресія додає штраф за абсолютне значення коефіцієнтів регресії до функції втрат. Це допомагає зменшити кількість ознак, що використовуються моделлю, шляхом відсічення неважливих ознак.
- **Формула:** $Loss = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^m |\beta_j|$
де:
 - α – параметр регуляризації, який контролює величину штрафу;
 - β_j – коефіцієнт регресії;
- **Механізм:** Lasso регресія додає штраф у вигляді суми абсолютних значень коефіцієнтів регресії. Це означає, що деякі коефіцієнти можуть стати нульовими, ефективно виключаючи відповідні ознаки з моделі.

- **Переваги:**
 - Виконує відсічення ознак, що спрощує модель і покращує інтерпретованість.
 - Допомогає запобігати перенавчанню.
- **Недоліки:**
 - Може призводити до високої варіації у випадках, коли існує висока кореляція між ознаками;

Оцінка моделі

Після навчання моделі лінійної регресії важливо оцінити її продуктивність. Ось деякі ключові метрики для оцінки моделі:

1. Середньоквадратична помилка (Mean Squared Error, MSE):

- **Визначення:** : Середньоквадратична помилка (MSE) є середнім значенням квадратів різниць між фактичними значеннями та передбаченими значеннями. Це міра середньої величини помилки для передбачень моделі.
- **Формула:**

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

де:

- n — кількість спостережень;
- y_i — фактичне значення;
- \hat{y}_i — передбачене значення;
- **Інтерпретація:**
 - Чим менше значення MSE, тим краща модель
 - MSE враховує великі помилки більше, ніж маленькі, оскільки помилки зводяться до квадрату.

2. Середня абсолютна помилка (MAE):

- **Визначення:** Середня абсолютна помилка (MAE) є середнім значенням абсолютних різниць між фактичними значеннями та передбаченими значеннями. Це міра середньої величини абсолютної помилки для передбачень моделі.
- **Формула:**

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

- **Інтерпретація:**

- Чим менше значення MAE, тим краща модель.
- MAE є більш інтерпретованою, оскільки виражена в тих же одиницях, що і залежна змінна.

3. Коефіцієнт детермінації (R^2):

- **Визначення:** Коефіцієнт детермінації R^2 показує, яка частка варіації залежної змінної пояснюється незалежними змінними моделі. Це міра того, наскільки добре модель пояснює варіацію в даних.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

де:

- \bar{y} – середнє значення залежної змінної;

- **Інтерпретація:**

- $R^2 = 1$: Модель ідеально пояснює дисперсію залежної змінної;
- $R^2 = 0$: Модель не пояснює дисперсію залежної змінної;
- Чим ближче значення R^2 до 1, тим краще модель пояснює дані;

Хід роботи

Завдання

1. Завантаження готових наборів даних з Scikit-learn:

- Завантажити набір `sklearn.datasets.fetch_california_housing`

2. Поділ даних на тренувальну та тестову вибірки:

- Поділити дані на тренувальний 70% та тестовий 30% набори.

3. Створити наступні регресійні моделі

- Поліноміальна регресія з ступенем поліному 2 та з регуляризацією: Ridge
- Поліноміальна регресія з ступенем поліному 20 та з регуляризацією: Ridge
- Поліноміальна регресія з ступенем поліному 2 та з регуляризацією: Lasso
- Поліноміальна регресія з ступенем поліному 20 та з регуляризацією: Lasso

4. Навчання та оцінка моделей:

- Для кожної створеної моделі виконати навчання виконати наступні операції

- a. Знайти оптимальний параметр регуляризації, який забезпечує найменшу середньоквадратичну помилку (похибку)
- b. Вивести коефіцієнти моделі
- c. Знайти оптимальний параметр регуляризації, який забезпечує найменшу середню абсолютну помилку (похибку)
- d. Вивести коефіцієнти моделі та порівняти з попередніми
 - Оцінити продуктивності моделі на тестових даних.
 - Результати оцінки похибок передбачення від коефіцієнта параметра регуляризації зобразити на графіках (середньоквадратична помилка та середня абсолютна помилка)

5. Оформити звіт

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import fetch_california_housing
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures, StandardScaler
from sklearn.linear_model import Ridge, Lasso
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.pipeline import make_pipeline

data = fetch_california_housing()
X, y = data.data, data.target

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

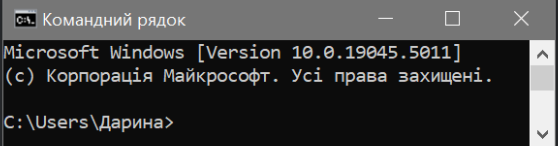


Рис. 1 Завантажування бібліотек та поділ тестових даних

```
def train_and_evaluate_model(model, degree, X_train, X_test, y_train, y_test, alpha_range):
    poly = PolynomialFeatures(degree=degree)
    mse_list = []
    mae_list = []

    for alpha in alpha_range:
        pipeline = make_pipeline(poly, StandardScaler(), model(alpha=alpha, max_iter=90000))
        pipeline.fit(X_train, y_train)

        y_pred = pipeline.predict(X_test)
        mse = mean_squared_error(y_test, y_pred)
        mae = mean_absolute_error(y_test, y_pred)
        mse_list.append(mse)
        mae_list.append(mae)

    optimal_alpha_mse = alpha_range[np.argmin(mse_list)]
    optimal_alpha_mae = alpha_range[np.argmin(mae_list)]

    best_model_mse = make_pipeline(poly, StandardScaler(), model(alpha=optimal_alpha_mse, max_iter=90000))
    best_model_mae = make_pipeline(poly, StandardScaler(), model(alpha=optimal_alpha_mae, max_iter=90000))

    best_model_mse.fit(X_train, y_train)
    best_model_mae.fit(X_train, y_train)
```

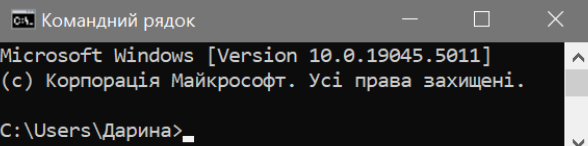


Рис. 2 Функція для тренування та оцінки моделі підбирає значення альфа для MAE/MSE. Усі дані для оцінювання зберігаються в масивах, а наприкінці результати зберігаються у словнику result.

```
print(f"{model.__name__} з поліномом ступеня {degree}:")
print(f" Найкращий alpha для MSE: {optimal_alpha_mse}")
print(f" Найкращий alpha для MAE: {optimal_alpha_mae}")
print(f" Коефіцієнти для MSE: {best_model_mse.named_steps[model.__name__].lower().coef_}")
print(f" Коефіцієнти для MAE: {best_model_mae.named_steps[model.__name__].lower().coef_}\n")

return (optimal_alpha_mse, best_model_mse.named_steps[model.__name__].lower().coef_,
        optimal_alpha_mae, best_model_mae.named_steps[model.__name__].lower().coef_,
        mse_list, mae_list)
```

Рис. 3 Продовження коду функції train_and_evaluate_model

```
alpha_range = np.logspace(-4, 4, 100)
degrees = [2, 6]
models = [(Ridge, 'Ridge'), (Lasso, 'Lasso')]

results = {}
for model, model_name in models:
    for degree in degrees:
        optimal_alpha_mse, coef_mse, optimal_alpha_mae, coef_mae, mse_list, mae_list = train_and_evaluate_model(
            model, degree, X_train, X_test, y_train, y_test, alpha_range)

        results[f'{model_name}_degree_{degree}'] = {
            'optimal_alpha_mse': optimal_alpha_mse,
            'coef_mse': coef_mse,
            'optimal_alpha_mae': optimal_alpha_mae,
            'coef_mae': coef_mae,
            'mse_list': mse_list,
            'mae_list': mae_list
        }

plt.figure(figsize=(10, 6))
plt.plot(alpha_range, mse_list, label="MSE", color='b')
plt.plot(alpha_range, mae_list, label="MAE", color='g')
plt.xscale('log')
plt.xlabel('alpha (параметр регуляризації)')
plt.ylabel('Помилка')
plt.title(f'{model_name} з поліномом ступеня {degree}')
plt.legend()
plt.show()
```

Рис. 4 Додавання даних оцінювання до словника 'result' та вивід графіків для моделей, які використовують регуляризацію Ridge (L2) та Lasso (L1) з поліномами степеня 2 та 6.

```

Ridge з поліномом ступеня 2:
Найкращий alpha для MSE: 0.026560877829466867
Найкращий alpha для MAE: 0.0001
Коефіцієнти для MSE: [ 0.          -15.95258302  -9.14696601   6.54351831  -4.87674561
-0.16188999 -0.16586811  7.73967362  1.54650267 -0.68688411
 0.1989768   0.74022836 -0.35402444  0.29918428 -0.60921716
-8.39069613 -25.44243555  0.18624079 -0.21954809  0.31147128
 0.07184096 -0.93538769 -3.95752162 -12.87826085  2.11803851
-3.54284789 -0.40431167  1.2293571  5.05133137 11.8863103
 1.47192836  0.6391616  -0.95783744 -4.22397533 -9.16533268
 0.03294554  1.75006248  0.53595494  1.0173704  0.7975893
 9.14268395 10.14826373  5.71624037 16.75379113  5.1207075 ]
Коефіцієнти для MAE: [ 0.00000000e+00 -2.23720300e+01 -1.01232279e+01  1.81054922e+01
-1.64793585e+01 -3.55692653e-01  1.08414773e+01  1.81464009e+01
 1.23435628e+01 -7.36374955e-01  1.34694865e-01  8.17039789e-01
-3.76772761e-01  2.70566566e-01 -4.48904148e-01 -1.09952713e+01
-3.45432153e+01  1.76391137e-01 -1.00464004e-01  2.05518253e-01
 6.52297016e-02 -9.52725784e-01 -4.47981770e+00 -1.44289470e+01
 2.30916133e+00 -4.24392771e+00 -3.53718373e-01  1.26633194e+00
 9.77096199e+00  2.79940628e+01  2.00820384e+00  5.76408769e-01
-1.10819185e+00 -9.13715981e+00 -2.53978524e+01  2.18693688e-02
 1.98375861e+00  4.38835428e-01  6.84889658e-01  6.94917004e-01
 9.63052845e+00  2.17915673e+01  9.08633911e+00  3.36775321e+01
 1.93559316e+01]

```

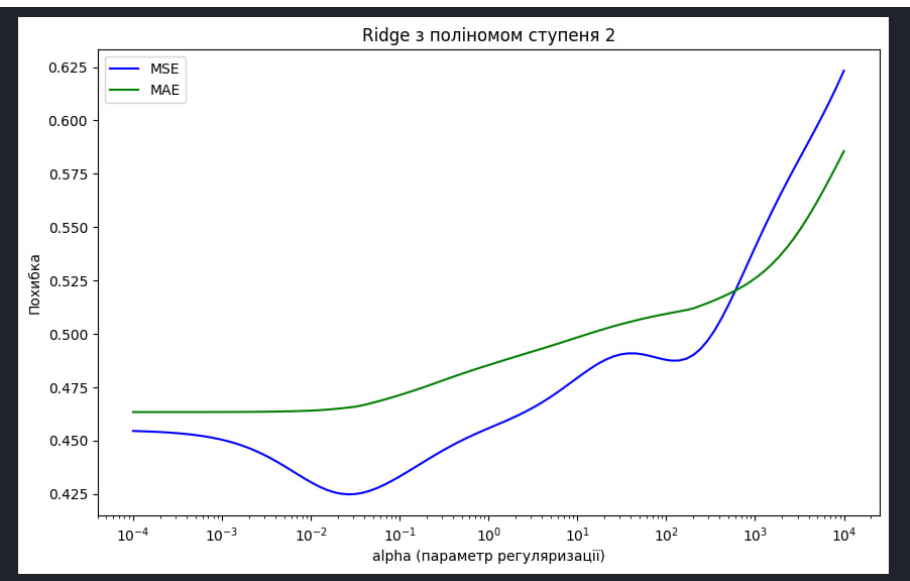


Рис. 5-6 Оцінка моделі з поліномом ступеня 2 за допомогою регуляризації Ridge показує графік, що ілюструє залежність похибки (MSE/MAE) від значення α . Високі значення α можуть призвести до недостатньої адаптації моделі, тоді як низькі значення можуть викликати перенасичення даними, що, в свою чергу, збільшує похибку.

Ridge з поліномом ступеня 6:

Найкращий alpha для MSE: 351.11917342151344

Найкращий alpha для MAE: 0.3593813663804629

Коефіцієнти для MSE: [0. -0.00038282 -0.01067336 ... -0.02151691 -0.01211223
0.10850409]

Коефіцієнти для MAE: [0. -0.65464618 -0.69392585 ... -0.93589061 0.26988733
1.1360515]

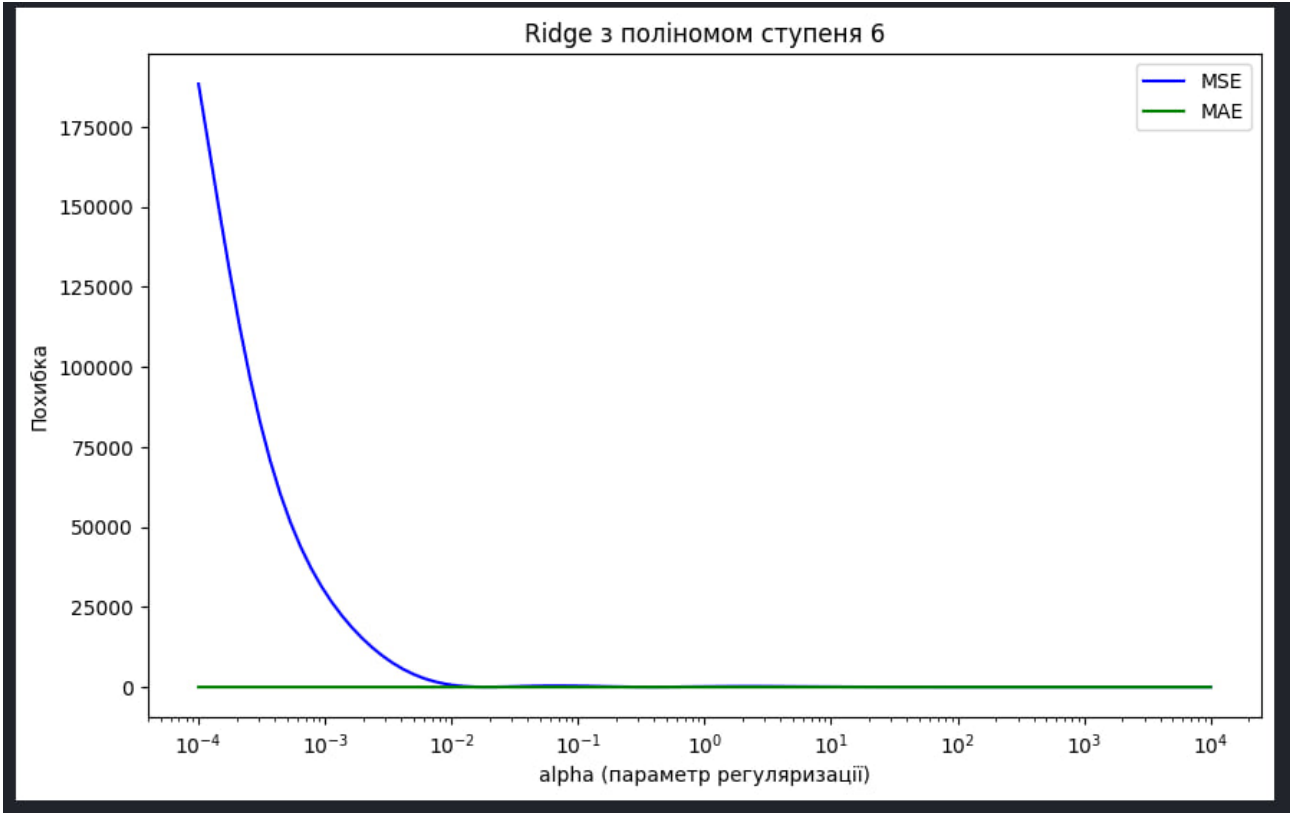


Рис. 7-8 Оцінка моделі поліному 6 ступеня з використанням Ridge регуляризації

```
Lasso з поліномом ступеня 2:
Найкращий alpha для MSE: 0.0001
Найкращий alpha для MAE: 0.0001
Коефіцієнти для MSE: [ 0.          -1.136304   -0.79846105 -0.          0.40648684 -0.47431971
-0.13860704 -0.08841477 -1.03984597 -0.5891007   0.35071174  0.68401407
-0.41387344  0.33536878 -0.67389436 -1.81356626 -3.92244556  0.22023225
-0.40471821  0.40898476  0.08012217 -1.09875285 -0.68643004 -1.18169582
 0.53848133 -0.4524913  -0.45190023  0.77458568  0.78706782  1.15429234
-0.06840414  0.71261237  0.24490667 -0.1549454   -0.          0.04558665
 1.36829402  0.40400218  0.63506893  1.2414145   5.91752185  7.55387743
 1.65821861  2.95537055  0.38431413]
Коефіцієнти для MAE: [ 0.          -1.136304   -0.79846105 -0.          0.40648684 -0.47431971
-0.13860704 -0.08841477 -1.03984597 -0.5891007   0.35071174  0.68401407
-0.41387344  0.33536878 -0.67389436 -1.81356626 -3.92244556  0.22023225
-0.40471821  0.40898476  0.08012217 -1.09875285 -0.68643004 -1.18169582
 0.53848133 -0.4524913  -0.45190023  0.77458568  0.78706782  1.15429234
-0.06840414  0.71261237  0.24490667 -0.1549454   -0.          0.04558665
 1.36829402  0.40400218  0.63506893  1.2414145   5.91752185  7.55387743
 1.65821861  2.95537055  0.38431413]
```

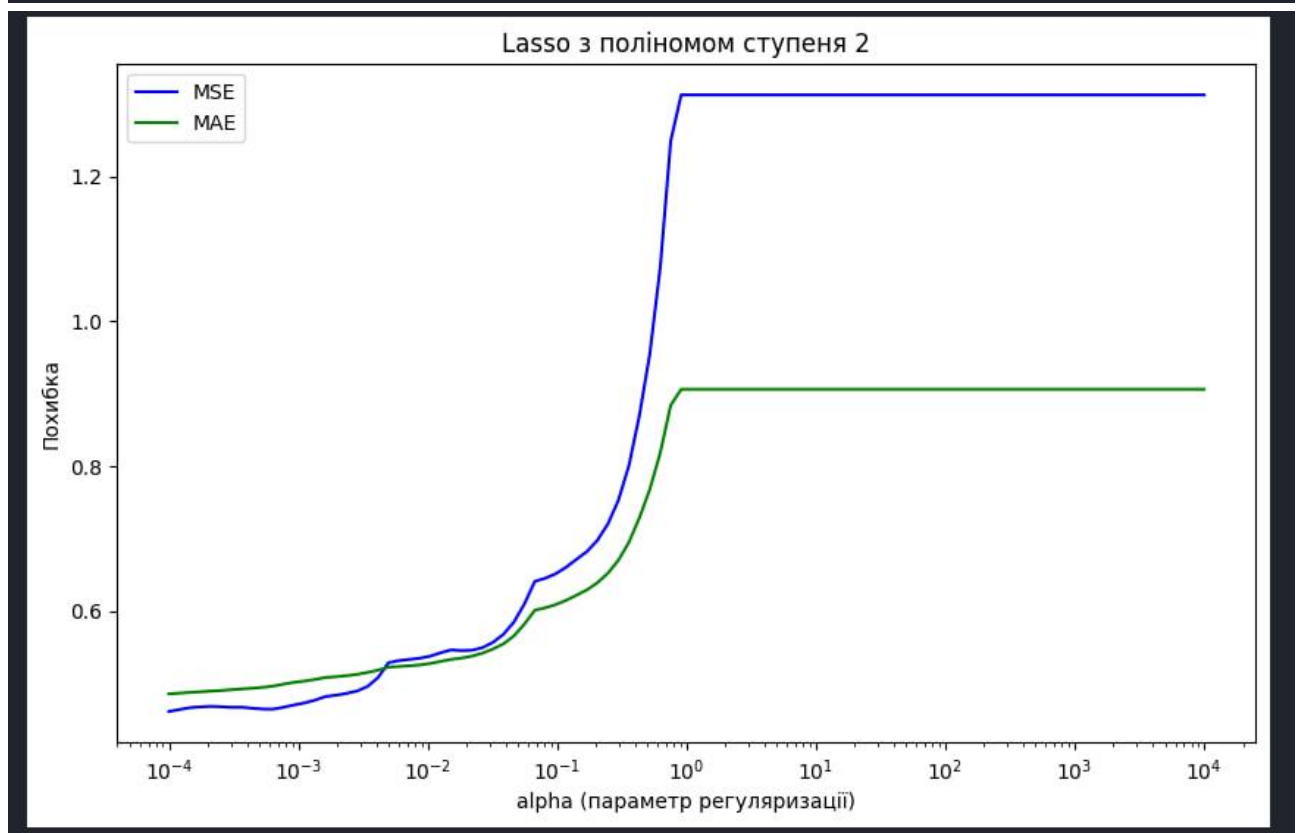


Рис. 9-10 Оцінка моделі поліному 2 ступеня з використанням Lasso регуляризації

Висновок: У цій лабораторній роботі я навчилася працювати з методами регуляризації L1/L2 (Lasso та Ridge). На основі виконаних завдань та отриманих результатів я помітила, що значення alpha безпосередньо впливає на MSE/MAE при низьких та високих значеннях. Зокрема, низькі значення alpha призводять до перенасичення даними, тоді як високі значення викликають їх нестачу, що суттєво збільшує похибку.