УДК 004.85 DOI: 10.18372/2073-4751.78.18968

Струк М.В.,
e-mail: maria.struk.02@gmail.com,
Моденов Ю.Б., к.т.н.,
orcid.org/0000-0003-3898-4159,
e-mail: yurii.modenov@npp.nau.edu.ua

ПЕРЕНАВЧАННЯ У СФЕРІ МАШИННОГО НАВЧАННЯ

Національний авіаційний університет

Вступ

Перенавчання є фундаментальною проблемою у сфері машинного навчання та статистичного моделювання, де модель добре працює на навчальних даних, але здатність узагальнювати нові, невідомі дані значно погіршується. В причинах перенавчання, лежать математичні основи, які забезпечують розуміння того, чому це відбувається та як впливає на продуктивність передбачуваних моделей.

Аналіз досліджень і публікацій

Останні дослідження в галузі машинного навчання пропонують різні стратегії для її розв'язання. Багато робіт зосереджуються на застосуванні різних методів регуляризації, таких як *dropout, L*1 та *L*2, для зменшення перенавчання. Інші досліджують використання ансамблевих методів, а також розвиток нових архітектур нейронних мереж, які менше схильні до переобладнання [2].

Невирішені аспекти проблеми

Одним з невирішених аспектів є знаходження оптимального балансу між складністю моделі та узагальненням даних. Іншими словами, як забезпечити, щоб модель була достатньо навченою для точного представлення складних взаємозв'язків у даних, але при цьому не була занадто складною, щоб уникнути перенавчання.

Тому ***метою*** даного дослідження є розгляд проблеми перенавчання в машинному навчанні. Зокрема, визначення різних математичних аспектів, що лежать в основі проблеми перенавчання, аналіз останніх дослідження та методів, які використовуються для її розв'язання, а також визначення невирішених аспектів цієї проблеми, на які слід звернути увагу у майбутніх дослідженнях.

Основний матеріал

Математична складність і ємність моделі. Основна причина надмірного оснащення часто полягає в складності моделі. Математично цю складність можна зрозуміти з точки зору кількості параметрів, які має модель. Наприклад, у поліноміальній регресії поліном вищого ступеня означає більше коефіцієнтів, що збільшує здатність моделі відповідати навчальним даним. Основне поняття у розумінні перенавчання – це компроміс зсуву та дисперсії. Великий зсув може призвести до того, що модель втратить релевантні зв'язки між функціями та цільовими результатами (недонавчання). Дисперсія, з іншого боку, це чутливість моделі до коливань у навчальному наборі. Велика дисперсія може спричинити перенавчання. Компроміс – це конфлікт між цими двома параметрами; зменшення одного зазвичай збільшує другий [2].

Статистична теорія навчання

1. ВЧ-розмірність: Розмірність Вапника–Червоненкіса (ВЧ) – це міра потужності (складності) алгоритму статистичної класифікації, як потужність найбільшої множини точок, яку цей алгоритм може роздрібнити. Модель з високою ВЧ-розмірністю, швидше за все, переповниться, оскільки вона може захоплювати більш складні шаблони (включаючи шум) у навчальних даних.

2. Регуляризація: регуляризація накладає обмеження на модель під час процесу навчання. Ці обмеження перешкоджають тому, щоб модель стала надто складною та заплутаною, що часто є

основною причиною перенавчання. Спрощуючи модель у контрольований спосіб, регулярізація гарантує, що вона фіксує справжні закономірності та зв'язки, властиві даним, підвищуючи її здатність до узагальнення [3].

Розмірність і кількість даних. Зі збільшенням кількості функцій або параметрів експоненціально зростає кількість даних, необхідних для точного узагальнення. У просторах великої розмірності точки даних розріджені, і модель, швидше за все, буде перенавчена через помилкові кореляції [2].

Співвідношення кількості параметрів у моделі до кількості спостережень (розміру вибірки) має вирішальне значення. Малий розмір вибірки з великою кількістю параметрів призводить до перенавчання. Статистично це може бути пов'язано зі ступенями свободи в моделі, де більше параметрів, ніж необхідно, з огляду на дані, може призвести до ідеальної відповідності навчальним даним, але поганого узагальнення.

Концепція складності моделі в контексті статистики та машинного навчання не має єдиного остаточного рішення, але її можна кількісно визначити або представити різними способами залежно від контексту та типу моделі, що використовується. Ось деякі з поширених способів математичного представлення або зв'язку зі складністю моделі:

1. Поліноміальна регресія: у поліноміальній регресії складність моделі зростає зі ступенем полінома. Для полінома ступеня d рівняння зазвичай представлено у вигляді (1):

$$y = \beta_0 + \beta_1 x + \beta_2 x^x + \ldots + \beta_d x^d. \quad (1)$$

Тут складність безпосередньо пов'язана з d , найвищим ступенем полінома.

2. Умови регуляризації [4]: у методах регуляризації, таких як Рідж і Ласо, складність моделі контролюється штрафним терміном, доданим до функції втрат. Для регресії Ріджа (регуляризація $L2$) член складності у функції втрат можна представити як (2):

$$Loss = MSE + \lambda \sum_{i=1}^{n} \beta_i^2. \quad (2)$$

А для регресії Лассо (регуляризація $L1$) (3):

$$Loss = MSE + \lambda \sum_{j=1}^{p} |\beta_j|, \quad (3)$$

де, *MSE* – середньоквадратична помилка, λ – параметр, який визначає силу регуляризації, n та p – кількість функцій або параметрів у моделі, а β_i та β_j –коефіцієнти, що відповідає i-тій та j-тій ознаці.

3. ВЧ-розмірність: Для моделей класифікації розмірність Вапника-Червоненкіса (ВЧ) є теоретичною мірою складності моделі. У ньому немає конкретного рівняння для обчислення, це більше концепція, яка використовується для опису здатності моделі класифікувати різні набори точок.

4. Кількість параметрів [5]: у багатьох моделях, особливо нейронних мережах, складність може бути приблизно оцінена кількістю параметрів, які можна навчити. Для нейронної мережі це можна представити як суму ваг і зміщень на всіх рівнях.

5. Інформаційні критерії : такі критерії, як інформаційний критерій Акаіке (*AIC*) і Баєсівінформаційний критерій (*BIC*), також визначають складність моделі. Вони щтрафують функцію ймовірності на основі кількості параметрів. Наприклад, *AIC* визначається як (4):

$$AIC = 2k - 2ln\,(\hat{L}), \quad (4)$$

де, k – кількість параметрів, а $\hat{L}$ – максимальне значення функції ймовірності для моделі.

Кожне з цих уявлень або заходів дає різний погляд на складність моделі та використовується в різних контекстах. Проте всі вони поділяють спільну тему балансу між відповідністю моделі даним (її точністю) та її простотою (щоб уникнути перенавчання).

Щоб продемонструвати причини перенавчання за допомогою математичної основи в *Python*, створено симуляцію, яка ілюструє ключові поняття, такі як складність моделі, вплив розміру вибірки,

компроміс зсуву та дисперсії. Використано поліноміальну регресію як приклад, оскільки це чіткий спосіб візуалізації перенавчання.

Спочатку окреслимо кроки, які використовуються в коді:

1. Згенерувати синтетичні дані: створити набір даних, який слідує за відомою функцією з деяким додаванням шуму.

2. Підгонка поліноміальних моделей: підгонка поліноміальних регресійних моделей різного ступеня (1, 4 і 15) до цього набору даних. Ступінь 1: Це простий лінійний поліном, де залежність між змінною впливу і відгуку моделюється лінійною функцією. Ступінь 4: Це поліном четвертого ступеня, де залежність моделюється квадратичною функцією з додаванням додаткових членів, що відображають нелинійні залежності. Ступінь 15: Це поліном п'ятнадцятого ступеня, який дозволяє моделювати дуже складні нелинійні залежності між змінними. Така модель може бути дуже гнучкою і може досить точно підганятися до даних, але також може виявитися дуже чутливою до шуму та перенавчання.

3. Оцінка продуктивність моделі: порівняння продуктивності цих моделей як на навчальному наборі, так і на тестовому наборі.

Розроблено код *Python* для цієї демонстрації:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.pipeline import make_pipeline

# Крок 1: Створення синтетичних даних
np.random.seed(0)
x = np.random.rand(100, 1) * 10  # Випадкові дані
y = np.sin(x) + np.random.randn(100, 1) * 0.5  # Синусоїдальний зв'язок із шумом

# Розділення даних на навчальні та тестові набори
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=0)

# Крок 2: Підбір поліноміальних моделей різного ступеня
degrees = [1, 4, 15]  # Степені полінома, який потрібно перевірити
plt.figure(figsize=(15, 5))

for i in range(len(degrees)):
    ax = plt.subplot(1, len(degrees), i + 1)
    plt.setp(ax, xticks=(), yticks=())

    polynomial_features = PolynomialFeatures(degree=degrees[i], include_bias=False)
    linear_regression = LinearRegression()
    pipeline = make_pipeline(polynomial_features, linear_regression)
    pipeline.fit(x_train, y_train)

    # Крок 3: Оцінення продуктивність моделі
    y_train_predict = pipeline.predict(x_train)
    y_test_predict = pipeline.predict(x_test)
    train_error = mean_squared_error(y_train, y_train_predict)
    test_error = mean_squared_error(y_test, y_test_predict)

    # Побудова графіків
    plt.plot(x_test, y_test_predict, label="Модель")
    plt.scatter(x_train, y_train, edgecolor='b', s=20, label="Дані")
    plt.xlabel("x")
    plt.ylabel("y")
    plt.xlim((0, 10))
    plt.ylim((-2, 2))
    plt.legend(loc="best")
    plt.title(f"Ступінь {degrees[i]}\nПНН: {train_error:.2f}, ПТН: {test_error:.2f}")
plt.show()
```

Результатом виконання коду є візуалізація підгонки зображена на рис. 1, щоб зрозуміти, як збільшення ступеня

полінома (збільшення складності моделі) призводить до перенавчання, на що вказує менша похибка в навчальному наборі, в той же час більша похибка в тестовому наборі для дуже складних моделей.

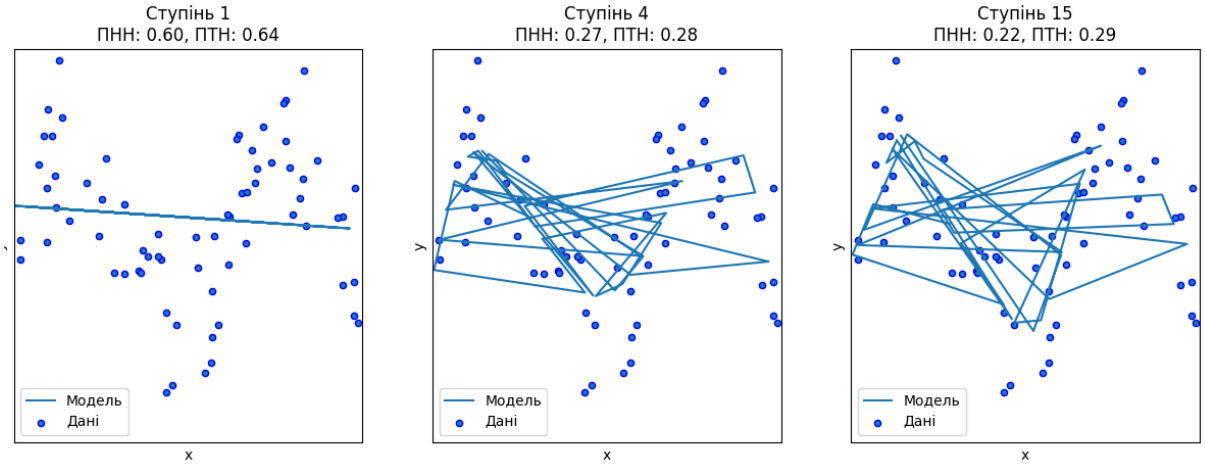

Рис. 1. Результат виконання коду

Висновки

Підсумовуючи, перенавчання – це багатогранна проблема, яка бере свій початок у математичних основах статистичного навчання. Це виникає через складну взаємодію між складністю моделі, розміром вибірки, компромісом зсуву та дисперсії. Співвідношення цих математичних концепцій має вирішальне значення для розробки надійних моделей, які добре узагальнюють нові, невідомі дані. Завдання для практиків полягає в тому, щоб збалансувати ці аспекти, гарантуючи, що моделі не будуть ані надто простими, щоб охопити основні закономірності, ані надто складними, щоб на них впливав шум у даних.

Література

1. What is overfitting?. URL: https://www.ibm.com/topics/overfitting?source=post_page-----09af234e9ce4---------- (дата звернення: 21.04.2024).

2. Fang C. et al. 4 – The Overfitting Iceberg. URL: https://blog.ml.cmu.edu/2020/08/31/4-overfitting/ (дата звернення: 26.04.2024).

3. Dijkinga F. J. Explaining L1 and L2 regularization in machine learning. URL: https://medium.com/@fernando.dijkinga/explaining-l1-and-l2-regularization-in-machine-learning-2356ee91c8e3 (дата звернення: 26.04.2024).

4. Oppermann A. Regularization in Deep Learning – L1, L2, and Dropout. URL: https://towardsdatascience.com/regularization-in-deep-learning-l1-l2-and-dropout-377e75acc036 (дата звернення: 25.04.2024).

5. Vignesh Sh. The Perfect Fit for a DNN. URL: https://medium.com/analytics-vidhya/the-perfect-fit-for-a-dnn-596954c9ea39 (дата звернення: 26.04.2024).

Струк М.В., Моденов Ю.Б.

ПЕРЕНАВЧАННЯ У СФЕРІ МАШИННОГО НАВЧАННЯ

Проблема перенавчання в машинному навчанні є актуальною та важливою для досягнення високої точності та надійності прогнозування на реальних даних. Ця стаття присвячена розгляду проблеми перенавчання з математичної перспективи. Вона починається з загального огляду проблеми та її важливості для наукових та практичних завдань, таких як розпізнавання образів, прогнозування та діагностика. Починаючи з

визначення ключових понять, таких як складність моделі, розмір вибірки, компроміс зсуву та дисперсії, текст розкриває взаємозв'язок між ними та вплив розміру вибірки на процес навчання. Для демонстрації цих концепцій розроблений код на мові програмування Python, який використовує поліноміальну регресію як модель для аналізу. Через створення синтетичних даних та підгонку різних моделей до них, ілюструється явище перенавчання та його вплив на точність прогнозів. Завершальні висновки наголошують на важливості розуміння математичних аспектів перенавчання для розробки надійних та ефективних моделей у машинному навчанні. Подальший аналіз останніх досліджень і публікацій у цій галузі демонструє різноманітні підходи до розв'язання проблеми, включаючи методи регуляризації, використання ансамблевих методів та розвиток нових архітектур нейронних мереж. Виокремлені невирішені аспекти, такі як знаходження оптимального балансу між складністю моделі та загальністю, які потребують подальшого дослідження. Остаточною метою статті є визначення ключових аспектів проблеми перенавчання та формулювання цілей для подальших досліджень в цій області.

***Ключові слова**: перенавчання; регуляризація (dropout, L1, L2); компроміс зсуву та дисперсії; поліноміальна регресія; ВЧ-розмірність.*

Struk M.V., Modenov Yu.B.

OVERFITTING IN MACHINE LEARNING

The problem of overfitting in machine learning is relevant and important for achieving high accuracy and reliability of predictions on real data. This article is dedicated to exploring the problem of overfitting from a mathematical perspective. It begins with a general overview of the problem and its importance for scientific and practical tasks such as pattern recognition, forecasting, and diagnostics. Starting with defining key concepts such as model complexity, sample size, bias-variance tradeoff, and dispersion, the text reveals the relationship between them and the influence of sample size on the learning process. To demonstrate these concepts, Python code is developed that uses polynomial regression as a model for analysis. Through the creation of synthetic data and fitting different models to them, the phenomenon of overfitting and its impact on prediction accuracy is illustrated. The concluding remarks emphasize the importance of understanding the mathematical aspects of overfitting for developing reliable and effective models in machine learning. Further analysis of recent research and publications in this field demonstrates various approaches to solving the problem, including regularization methods, the use of ensemble methods, and the development of new neural network architectures. Unresolved aspects, such as finding the optimal balance between model complexity and generality, are highlighted for further investigation. The ultimate goal of the article is to identify the key aspects of the overfitting problem and formulate goals for further research in this area.

***Keywords**: overfitting; regularization (dropout, L1, L2); bias-variance tradeoff; polynomial regression; VC dimension.*