

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет

Факультет Електроніки та інформаційних технологій

Кафедра прикладної математики та моделювання складних систем

«До захисту допущено»

Завідувач кафедри

_____ Ігор КОПЛИК

_____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня магістр

зі спеціальності 113 «Прикладна математика» ,

освітньо-професійної програми «Наука про дані та моделювання складних систем»
на тему: «Моделювання процесу зносу ріжучих інструментів з захисними покриттями
за допомогою методів машинного навчання»

Здобувача (ки) групи ПМ.м-31 Негреби Євгенія Юрійовича

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело.

_____ Євгеній НЕГРЕБА

Керівник професор, доктор фіз.-мат. наук, Олександр ГОНЧАРОВ _____

Суми – 2024

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Факультет	електроніки та інформаційних технологій	
Кафедра	прикладної математики та моделювання складних систем	
Рівень вищої освіти	другий (магістр)	
Галузь знань	11 Математика та статистика	
Спеціальність	113 Прикладна математика	
Освітня програма	освітньо-професійна	«Наука про дані та моделювання складних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри ПМ та МСС

Коплик І.В. _____

«__» _____ 2024 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Негребі Євгенію Юрійовичу

- Тема роботи Моделювання процесу зносу ріжучих інструментів з захисними покриттями за допомогою методів машинного навчання/ Modeling the wear process of cutting tools with protective coatings using machine learning methods.
Керівник роботи Гончаров О.А., професор, доктор фіз.-мат. наук
затверджено наказом по факультету ЕлІТ від «__» _____ 202_ р. № _____
- Термін подання роботи студентом «1» грудня 2024 р.
- Вихідні дані до роботи 1) Теоретичні відомості про процес зносу різця на основі аналізу даних; 2) Класифікаційні методи машинного навчання: метод випадкових лісів та метод найближчих сусідів;
- Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) Дослідження факторів, що впливають на знос ріжучих інструментів. Побудова математичної моделі процесу зносу на основі аналізу згенерованих даних з використанням методів машинного навчання. Практична реалізація побудованої математичної моделі.

5. Перелік графічного матеріалу:

- 1) Крива зносу інструменту як функція швидкості різання і згенеровані дані;
- 2) Візуалізація класифікації методом випадкових лісів;
- 3) Візуалізація класифікації методом найближчих сусідів;
- 4) Графік порівняння середнього терміну служби інструменту для різних типів покриттів.

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «21» жовтня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування роботи, заходи	Термін виконання	Примітка
1	Провести аналіз літератури за темою «Моделювання процесу зносу ріжучих інструментів з захисними покриттями за допомогою методів машинного навчання.».	21.10-28.10	Виконано
2	Дослідити фактори, що впливають на знос ріжучих інструментів.	29.10-09.11	Виконано
3	Побудувати математичну модель процесу зносу на основі аналізу даних з використанням методів машинного навчання.	10.11-21.11	Виконано
4	На базі математичної моделі створити комп'ютерну програму.	22.11-01.12	Виконано

Здобувач вищої освіти

(підпис)

Негреба Є.Ю.

Керівник роботи

(підпис)

Гончаров О.А.

РЕФЕРАТ

Кваліфікаційна робота: 47 с., 12 рис., 5 табл., 42 джерела.

Мета роботи: Моделювання процесу зносу ріжучих інструментів із захисними покриттями та порівняльний аналіз ефективності захисних покриттів для продовження терміну служби інструментів.

Об'єкт дослідження: Процес зношування ріжучих інструментів без захисного покриття, з одношаровим та багатшаровими покриттями при механічній обробці.

Предмет дослідження: Комп'ютерна модель зносу ріжучих інструментів для прогнозування терміну служби в металевій пластині без захисного покриття та з різними типами захисних покриттів.

Методи дослідження: метод генерування синтетичних даних для моделювання процесу зношування, методи машинного навчання (випадкові ліси, метод найближчих сусідів), методи балансування класів, аналітичний аналіз.

В результаті досліджень створено комп'ютерну модель на основі методів машинного навчання, яка прогнозує термін служби ріжучих інструментів залежно від типу захисного покриття. Модель дозволяє порівняти середній термін служби для різних покриттів та визначити найбільш ефективне покриття для зменшення зносу й підвищення продуктивності.

КЛЮЧОВІ СЛОВА: МАТЕМАТИЧНА МОДЕЛЬ, ЗАХИСНЕ ПОКРИТТЯ, РІВНЯННЯ ТЕЙЛОРА, ГЕНЕРАЦІЯ ДАНИХ, МАШИННЕ НАВЧАННЯ, КЛАСИФІКАЦІЯ, МЕТОД ВИПАДКОВИХ ЛІСІВ, МЕТОД НАЙБЛИЖЧИХ СУСІДІВ, БАЛАНСУВАННЯ КЛАСІВ, СЕРЕДНІЙ ТЕРМІН СЛУЖБИ.

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1	8
ЛІТЕРАТУРНИЙ ОГЛЯД	8
Висновки до розділу 1	14
РОЗДІЛ 2	16
МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧІ	16
2.1 Основне рівняння Тейлора	16
2.2 Врахування стохастичних факторів	17
2.3 Генерація даних зносу	17
2.4 Генерація даних для різних покриттів	18
2.5 Класифікаційна модель на основі отриманих даних	18
2.5.1 Метод випадкового лісу (Random Forest)	19
2.5.2 Метод найближчих сусідів(K-Nearest Neighbors)	21
2.6 Балансування даних і обчислення точності моделі	22
2.7 Розрахунок середнього терміну служби на основі класифікації	24
Висновки до розділу 2	25
РОЗДІЛ 3	27
РЕАЛІЗАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ	27
3.1 Генерація даних про знос інструменту	28
3.2. Методи машинного навчання для класифікації зносу інструменту	30
3.2.1 Метод випадкових лісів(Random Forest)	30
3.2.2 Метод найближчих сусідів(KNN)	31
3.3 Аналіз середньої тривалості служби на основі кращої моделі	33
ВИСНОВКИ	34
СПИСОК ВИКРИСТАНИХ ДЖЕРЕЛ	35
ДОДАТОК А	40

ВСТУП

Знос інструменту при механічній обробці (фрезерування, токарна обробка, свердління) визначається як втрата матеріалу з ріжучої кромки інструменту під час роботи через взаємодію з матеріалом заготовки [1]. Знос інструменту є процесом, що залежить від температури і від параметрів обробки (швидкість різання, подача, глибина різання), властивостей матеріалу (твердість, теплопровідність), властивостей охолоджуючої рідини та способу її подачі, а також від властивостей інструменту (тип, геометрія, покриття, і товщина). Термін служби інструменту визначається як час, за який знос інструменту досягає попередньо встановленої межі зносу [1]. Межа зносу встановлюється тоді, коли вона стає достатньо великою, щоб бути обмежувальним фактором для процесу через зниження якості деталі або ризик катастрофічного руйнування інструменту, що може призвести до браку або переробки деталі.

У літературі було багато спроб моделювання та прогнозування терміну служби інструменту. Моделі є емпіричними, такими як рівняння терміну служби інструменту Тейлора або методологія поверхні відгуку [1-6], або фізично-обґрунтованими моделями з використанням температурних методів або методів кінцевих елементів [7-10]. Проблемою практичної реалізації існуючих моделей є те, що вони вимагають проведення широких експериментів зі зносу інструменту для калібрування коефіцієнтів моделі, що є дорогим і трудомістким для різних комбінацій інструмент-матеріал, і тому є нездійсненним у промислових умовах. Крім того, знос інструменту є стохастичним через варіації у роботі окремих інструментів [11]. Як наслідок, моделювання та прогнозування терміну служби інструменту вважається складним для передбачення.

Основна задача даної роботи — моделювання терміну служби для 4 типів ріжучих інструментів за допомогою різних методів машинного навчання на основі даних про знос, зібраних у виробничих цехах. Досліджувалися ріжучі інструменти без покриття, з одношаровим покриттям TiN, двошаровим TiN/ α -

Al₂O₃ та тришаровим TiN/ α -Al₂O₃/TiCN. У процесі моделювання встановлені терміни служби для всіх типів інструментів, що дозволить обрати найбільш вигідний варіант.

РОЗДІЛ 1

ЛІТЕРАТУРНИЙ ОГЛЯД

Знос ріжучого інструменту обмежує продуктивність обробки. Одним з ефективних шляхів підвищення працездатності різального інструменту являється нанесення на його робочі поверхні зносостійких захисних покриттів. Для того щоб оцінити ефективність використання захисних покриттів, прогнозування життєвого циклу ріжучих інструментів є критично важливим. Зношування інструменту залежить від параметрів обробки (швидкість різання, швидкість подачі та глибина різання), властивостей матеріалу (твердість, теплопровідність), а також властивості інструменту (тип, геометрія, захисні покриття та товщина). Традиційні емпіричні та фізичні моделі потребують експериментального калібрування, яке є неможливим в умовах великої кількості комбінацій систем інструмент-заготовка. Застосування методів машинного навчання для моделювання процесу зносу ріжучих інструментів може значно підвищити точність прогнозів та ефективність виробничих процесів.

В дослідженні[12] виготовлення деталей на виробництві розглядається, як експерименти щодо зносу інструменту. Для моделювання ресурсу інструменту використовувалися дані про знос інструменту. Але основна проблема використання даних виробництва полягає в тому, що знос інструменту можна виміряти лише під час заміни інструменту, що призводить до однієї точки даних разом із кривою зносу інструменту від часу різання. Щоб вирішити цю проблему, довговічність інструменту розглядалася як проблема класифікації на відміну від проблеми регресії. Варто зазначити, що термін служби інструменту визначається як час, необхідний для того, щоб знос інструмента досягнув попередньо визначеної межі зносу.

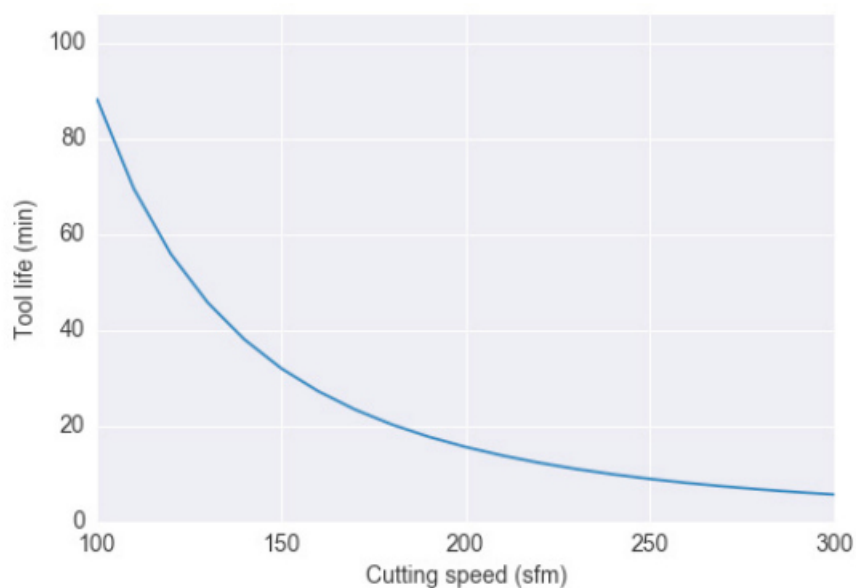


Рис.1.1 Істинна крива довговічності інструмента як функція швидкості різання[12]

Інструмент не вийшов з ладу, якщо знос інструменту менший за порогове значення, і інструмент вийшов з ладу, якщо він перевищує порогове значення.

Було визначено два класи; клас 0 означає, що інструмент не вийшов з ладу, а клас 1 означає, що інструмент вийшов з ладу. Стійкість інструменту — це межа класифікаційного рішення в часі, коли прогнозований клас інструменту змінюється від 0 до 1.

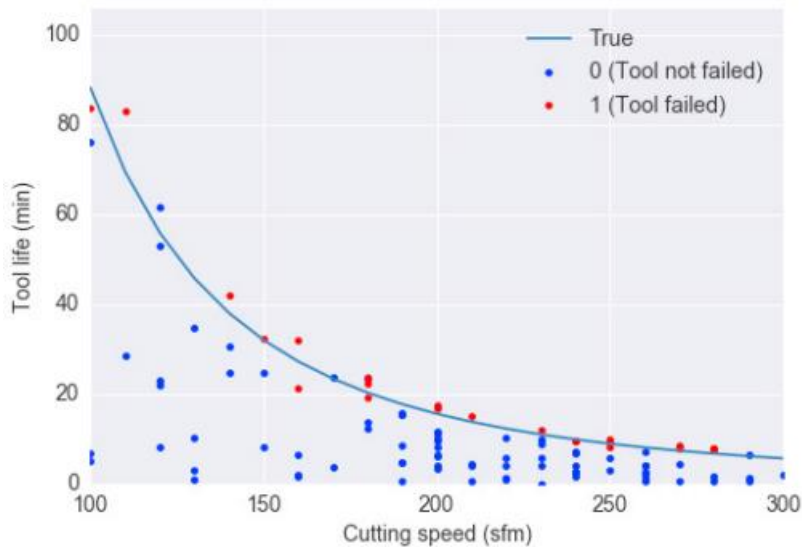


Рис. 1.2 Істинна крива довговічності інструменту як функція швидкості різання зі 100 випадковими змодельованими точками даних; синій означає, що інструмент не зношений (клас 0), а червоний означає, що інструмент вийшов з ладу (клас 1) [12]

Основним внеском даного дослідження є демонстрація ефективності різних методів машинного навчання для прогнозування терміну служби інструменту з використанням змодельованих даних про знос інструменту.

Машинне навчання визначається як область дослідження, яка дає комп'ютерам можливість навчатися без явного програмування [13]. Більш формально машинне навчання також визначається так: «кажуть, що комп'ютерна програма вчиться на досвіді E щодо деякого класу завдань T і показника продуктивності P , якщо її продуктивність у виконанні завдань у T , виміряну P , покращується з досвідом. E [14]. У випадку класифікації ресурсу інструменту досвід E отримується за допомогою даних про знос, завдання T полягає в тому, щоб класифікувати інструмент як справний або несправний, а міра ефективності P є точністю класифікації. Кожні дані про знос інструменту є окремою точкою на кривій зносу інструмента від часу різання та дають інформацію про те, чи інструмент вийшов з ладу чи ні. Тому довговічність інструменту розглядається як проблема класифікації. Оскільки довговічність інструменту як функція

швидкості різання безперервна і монотонно зменшується, для оцінки обрано два методи класифікації машинного навчання; метод опорних векторів (SVM) і логістичний метод.

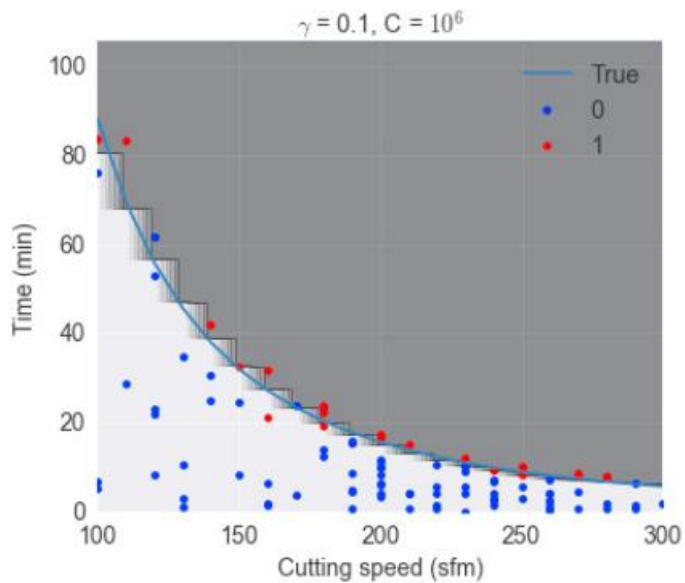


Рис. 1.3. Рішення методом RBF SVM $\gamma = 0,1, C = 10^6$. Біло-сіра область відповідає класу 0 і класу 1 передбачення відповідно. [12]

$n = 25$	$n = 50$	$n = 100$	$n = 500$
0.828	0.85	0.861	0.876

Табл.1.1 Точність класифікації для RBF SVM моделюванням Монте-Карло в залежності від кількості точок вибірки(n) ($C = 1 \times 10^6, \gamma = 0,1$). [12]

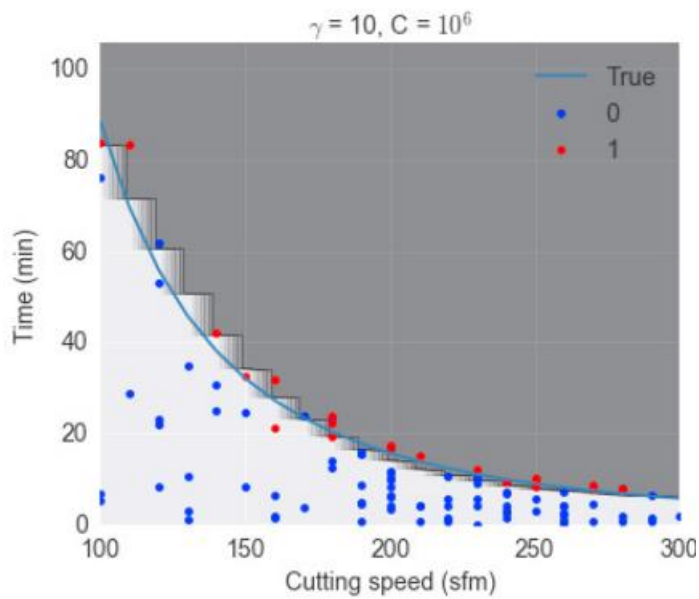


Рис. 1.4. Рішення методом поліноміального SVM $\gamma = 10$, $C = 10^6$. Біло-сіра область відповідає класу 0 і класу 1 передбачення відповідно. [12]

$n = 25$	$n = 50$	$n = 100$	$n = 500$
0.828	0.85	0.863	0.878

Табл.1.2 Точність класифікації для поліноміального SVM моделюванням Монте-Карло в залежності від кількості точок вибірки(n) ($C = 1 \times 10^6$, $\gamma = 10$). [12]

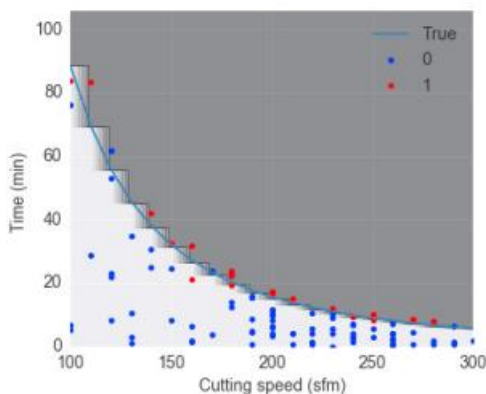


Рис. 1.5. Рішення методом логістичної класифікації з логарифмічним перетворенням ,при $C = 10^6$. Біло-сіра область відповідає класу 0 і класу 1 передбачення відповідно. [12]

$n = 25$	$n = 50$	$n = 100$	$n = 500$
0.842	0.860	0.874	0.880

Табл.1.3 Точність класифікації для логістичної класифікації моделюванням Монте-Карло в залежності від кількості точок вибірки(n) ($C = 10^6$). [12]

Для побудови прогнозованих моделей довговічності інструменту, окрім класичних методів машинного навчання, дослідники використовують моделі штучних нейронних мереж (ANN).

Гарний огляд контролю сили різання та моніторингу зносу інструменту при торцевому фрезеруванні можна знайти в [15]–[21].

В роботі[22] досліджували схему довгострокового контролю сил різання для регулювання довговічності інструменту в процесах торцевого фрезерування

Показники, засновані на силі фрезерування для зносу інструменту при фрезеруванні, були досліджені [23].

Вчені[24] досліджували оцінку зносу інструменту в режимі реального часу на основі сили різання при торцевому фрезеруванні з використанням комбінації методів обробки сигналів, тоді як знос інструменту при високошвидкісному фрезеруванні з використанням підходу процесу виявлення було виконано в роботі [25].

Оцінка зносу інструменту під час фрезерування з ЧПК за допомогою нейронної мережі(ANN) сенсорного синтезу була реалізована в роботі[26].

Онлайн-моніторинг стану металорізального інструменту за допомогою багат шарових нейронних мереж перцептронів вивчали Лістер і Дімла [27].

Моніторинг стану інструменту за допомогою методів штучного інтелекту здійснено в роботі [28].

Науковці оцінили довжину зносу інструменту при чистовому фрезеруванні за допомогою алгоритму нечіткого висновку [29].

Інтелектуальний нагляд за процесом для прогнозування зносу інструменту в процесах обробки був зроблений [30].

В роботі [31] вчені проаналізували моделювання механізму фрезерування кульовим торцем, включаючи знос інструменту. Прогнозування зношування бокової частини за допомогою моделювання нейронної мережі зворотного поширення під час різання загартованої сталі H-13 інструментами зі скошеними та відточеними CBN використовується в роботі [32].

Також реалізовано моделювання сили різання інструменту при кульовому фрезеруванні з використанням багаторівневого перцептрона.[33].

Вчені [34] працювали над виявленням зносу інструменту при сухому високошвидкісному фрезеруванні на основі аналізу внутрішніх сигналів машини, а група науковців [35] оцінювала знос інструменту за допомогою аналізу вібрації під час торцевого фрезерування холодної інструментальної сталі AISI D3 з твердістю 35 HRC.

Крім того, в роботі [36] досліджено вплив акустичного випромінювання та характеристик поверхні за допомогою штучної нейронної мережі.

Висновки до розділу 1

Літературний огляд показав, що застосування методів машинного навчання для прогнозування процесу зносу ріжучих інструментів є перспективним напрямком досліджень. Незважаючи на значні досягнення у застосуванні методів машинного навчання для прогнозування життєвого циклу

ріжучих інструментів, існують певні прогалини. По-перше, необхідно уніфікувати підходи до збору та аналізу даних, оскільки різні умови обробки можуть значно впливати на результати. По-друге, необхідні подальші дослідження щодо впливу різних умов експлуатації на ефективність захисних покриттів.

Майбутні дослідження можуть зосередитися на розробці гібридних моделей, що поєднують різні методи машинного навчання для підвищення точності прогнозування. Подальші дослідження в цій галузі допоможуть удосконалити методи прогнозування та знайти нові рішення для підвищення ефективності ріжучих інструментів. Крім того, перспективним напрямком є дослідження нових матеріалів покриття та їхнього впливу на довговічність інструментів.

РОЗДІЛ 2

МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧІ

Щоб вирішити проблему прогнозування ресурсу ріжучого інструменту в промислових умовах, у роботі представлено методи класифікації машинного навчання для моделювання ресурсу на основі даних про знос у виробничих цехах. Мета полягає в тому, щоб розглядати виробничі деталі як експерименти щодо зношування різця та використовувати ці дані для моделювання ресурсу різця. Основна проблема полягає в тому, що зношування можна виміряти лише під час заміни різця, що дає лише одну точку даних, яка пов'язана з кривою зносу за часом різання. Для вирішення цієї проблеми довговічність розглядалася як задача класифікації замість регресії. Термін служби визначається як час, необхідний для досягнення попередньо визначеної межі зносу. Якщо зношування менше за порогове значення — різець справний, якщо більше — вийшов з ладу.

Варто зазначити, що відмова інструменту не означає катастрофічної поломки. Було визначено два класи: клас 0 означає, що інструмент не вийшов з ладу, а клас 1 — що інструмент вийшов з ладу. Стійкість інструменту — це межа класифікаційного рішення в часі, коли прогнозований клас змінюється з 0 на 1.

Модель базується на емпіричному рівнянні Тейлора для прогнозування зносу інструменту, яке використовується для оцінки терміну служби інструменту. Це рівняння залежить від швидкості різання та інших параметрів інструменту і робочих умов. Метою моделі є передбачення часу досягнення інструментом граничного рівня зносу для різних конфігурацій покриттів.

2.1 Основне рівняння Тейлора

Використаємо рівняння Тейлора для оцінки терміну служби інструменту

$$vT^n = C \quad (1)$$

де v -швидкість різання в м/хв, T - термін служби інструменту (хв), n - показник, що відображає вплив швидкості різання на знос інструменту, C - константа, що залежить від матеріалу інструменту і умов обробки.

2.2 Врахування стохастичних факторів

Зношення інструменту є процесом зі стохастичним характером, тому до рівняння додаємо невизначеність. Припустимо, що термін служби T розподіляється за нормальним законом із середнім значенням, обчисленим за допомогою рівняння Тейлора(1), і стандартним відхиленням σ у вигляді 20% від середнього значення.

Тобто:

$$T \sim \mathcal{N}\left(\frac{C}{V^n}, \sigma\right) \quad (2)$$

де $\sigma = 0.2 \cdot \frac{C}{V^n}$.

2.3 Генерація даних зносу

Для моделювання зношення інструменту створюємо вибірки швидкості різання V з рівномірного інтервалу $[V_{min}, V_{max}]$, де $V_{min}=30$ м/хв, а $V_{max}=100$ м/хв.

Для кожного обраного значення V :

1. Розраховуємо середнє значення T за рівнянням (2).
2. Генеруємо термін служби T_{sample} за нормальним розподілом:

$$T_{sample} = T + \sigma \cdot Z \quad (3)$$

де $Z \sim \mathcal{N}(0, 1)$ — стандартне нормально розподілене випадкове значення.

3. Визначаємо реальний час різання t , при якому відбувається зношення, шляхом вибірки випадкового коефіцієнта r з інтервалу $[0,1.2]$:

$$t = r \cdot T \quad (4)$$

Якщо $t < T_{sample}$, інструмент не зношений (клас 0). Інакше інструмент вважається зношеним (клас 1).

2.4 Генерація даних для різних покриттів

Хоча параметри n та C з рівняння (1) визначаються експериментально, у відкритому доступі є табличні значення даних параметрів для різних типів інструментів. Параметри n та C для різних типів різця:

- без покриття(ISO P20):

$$n=0.4, C=183 [37]$$

- з одношаровим покриттям(ISO P20/ TiN):

$$n=0.41, C=198 [38]$$

- з двошаровим покриттям(ISO P20/ TiN/ α -Al₂O₃):

$$n=0.43, C=229 [39]$$

- з тришаровим покриттям(ISO P20/ TiN/ α -Al₂O₃/TiCN):

$$n=0.45, C=244 [40]$$

Таким чином, можна описати довговічність інструментів залежно від їх покриттів.

2.5 Класифікаційна модель на основі отриманих даних

Для прогнозування стану інструменту використовуємо методи класифікації, такі як випадковий ліс(Random Forest) та метод найближчих

сусідів(K-Nearest Neighbors). Дані методи обрані через їх високу точність, стійкість до шуму та здатність обробляти великі обсяги даних.

2.5.1 Метод випадкового лісу (Random Forest)

Випадковий ліс — це ансамблевий метод класифікації, який використовує безліч дерев рішень для підвищення загальної точності.[41]

1. Побудова дерева

1.1 Вибірка за методом бутстрепа: на першому етапі для кожного дерева виконується вибірка з поверненням з навчального набору даних.

Випадковим чином обирається $N_{samples}$ зразків із загального числа N .

1.2 Розбиття вузлів: кожен вузол розбивається на основі найкращої ознаки X_j та порогового значення θ , щоб мінімізувати критерій розбиття (індекс Джині). Розбиття виконується на дві підмножини: D_{left} і D_{right} .

Формально функція розбиття $h(x)$ на вузлі визначається як:

$$h(x) = \arg \min_{j, \theta} [\text{Gini}(D_{left}) + \text{Gini}(D_{right})] \quad (5)$$

де:

- D_{left} і D_{right} . — підмножини, отримані після розбиття вузла на основі ознаки X_j та порогового значення θ ,
- $\text{Gini}(D) = 1 - \sum_{k=1}^K p_k^2$ — індекс Джині для підмножини D , де p_k — частка зразків, що належать до k -го класу.

Таким чином, мінімізація суми значень індексу Джині для D_{left} і D_{right} (5) дозволяє вибрати оптимальне розбиття, яке зменшує змішаність класів у нових підмножинах.

2. Ансамбль дерев

Випадковий ліс створюється як ансамбль N_{trees} дерев рішень. Для кожного дерева застосовуються ті ж самі кроки побудови, але з унікальною бутстреп-вибіркою та випадковою підмножиною ознак для кожного розбиття вузла.

3. Прогноз ансамблю дерев

Прогноз для кожного об'єкта даних виконується шляхом голосування результатів усіх дерев у лісі:

$$\hat{y} = \text{majority vote } \{\hat{y}_{tree_i}\}_{i=1}^{N_{trees}} \quad (6)$$

де:

- \hat{y}_{tree_i} — прогноз i -го дерева в лісі,
- majority vote — функція, яка обирає клас із найбільшою кількістю голосів від усіх дерев.

Нехай:

- \hat{y}_{tree_i} — прогноз i -го дерева в лісі,
- N_{trees} — загальна кількість дерев у лісі,
- K — кількість класів.

Тоді majority vote для кожного об'єкта визначається так:

$$\hat{y} = \arg \max_{k \in \{1, \dots, K\}} \sum_{i=1}^{N_{trees}} \mathbb{I}(\hat{y}_{tree_i} = k) \quad (7)$$

де:

- $\mathbb{I}(\hat{y}_{tree_i} = k)$ — індикаторна функція, яка дорівнює 1, якщо $\hat{y}_{tree_i} = k$ (тобто дерево i прогнозує клас k), і 0 в іншому випадку,
- $\arg \max$ обирає клас k , для якого сума голосів є найбільшою.

2.5.2 Метод найближчих сусідів(K-Nearest Neighbors)

Метод K-Nearest Neighbors (KNN) класифікує новий зразок, обираючи k -найближчих сусідів у багатовимірному просторі ознак і присвоюючи зразку той клас, який має більшість серед цих сусідів.[42]

1. Визначення відстані між точками

Для визначення найближчих сусідів використовуємо метрику відстані між точками. У KNN найчастіше застосовується евклідова відстань. Евклідова відстань обчислює "прямолінійну" відстань між двома точками, що є найкоротшою відстанню в багатовимірному просторі. Для двох точок $x = (x_1, x_2, \dots, x_M)$ та $x' = (x'_1, x'_2, \dots, x'_M)$ у просторі ознак із M вимірами евклідова відстань визначається як:

$$d(x, x') = \sqrt{\sum_{j=1}^M (x_j - x'_j)^2} \quad (8)$$

де:

- x_j і x'_j — значення j -ї ознаки для точок x та x' відповідно,
- M — загальна кількість ознак.

2. Пошук k -найближчих сусідів

2.1 Обчислення відстаней: Для нового зразка x обчислюється відстань до кожного зразка в навчальному наборі.

2.2 Сортування: Всі зразки в навчальному наборі сортуються за зростанням відстані до x .

2.3 Вибір k -найближчих сусідів: Обираються k найближчих зразків, які мають мінімальну відстань до x .

3. Прогнозування класу

Для нового зразка x прогнозується клас, якому належить більшість з його k -найближчих сусідів.

Позначимо:

- N_0 — кількість сусідів класу 0 серед k -найближчих сусідів,
- N_1 — кількість сусідів класу 1 серед k -найближчих сусідів.

Прогнозований клас \hat{y} визначається як клас із найбільшою кількістю сусідів:

$$\hat{y} = \arg \max\{N_0, N_1\} \quad (9)$$

де:

- якщо $N_0 > N_1$, то $\hat{y} = 0$,
- якщо $N_1 > N_0$, то $\hat{y} = 1$.

У випадку рівної кількості сусідів (тобто $N_0 = N_1$) можна обрати клас випадковим чином або зменшити k до непарного значення).

2.6 Балансування даних і обчислення точності моделі

Для балансування класів (вирівнювання кількості об'єктів класів 0 та 1) застосовуємо метод генерації синтетичних даних, SMOTE (Synthetic Minority Over-sampling Technique).

SMOTE створює нові об'єкти меншості шляхом інтерполяції між вибраним об'єктом меншості x_i і одним із його k -найближчих сусідів $x_{neighbor}$:

1. Вибираємо об'єкт меншості x_i і випадкового сусіда $x_{neighbor}$.

2. Генеруємо новий об'єкт за формулою:

$$x_{\text{new}} = x_i + \lambda \cdot (x_{\text{neighbor}} - x_i) \quad (10)$$

де $\lambda \sim U(0,1)$ — випадкове число з інтервалу від 0 до 1.

Цей процес дозволяє утворювати нові об'єкти, зберігаючи структуру простору ознак, але водночас вирівнюючи кількість об'єктів обох класів.

Метод SMOTE (Synthetic Minority Over-sampling Technique) добре працює для методів, чутливих до розподілу даних, таких як KNN, оскільки він створює нові синтетичні об'єкти меншості, що допомагає збалансувати класи. Натомість Random Forest добре працює з дисбалансом класів завдяки можливості встановлювати параметр `class_weight`, що дозволяє збільшити вагу менших класів під час навчання, компенсуючи дисбаланс без створення додаткових даних.

Нехай:

- N — загальна кількість зразків у навчальній вибірці,
- N_0 — кількість зразків класу 0 ,
- N_1 — кількість зразків класу 1 .

Тоді коефіцієнти зважування класів визначаються за формулою:

$$\begin{aligned} \text{class_weight}_0 &= 1 \\ \text{class_weight}_1 &= \frac{N}{2 \cdot N_1} \end{aligned} \quad (11)$$

де:

- class_weight_0 — вага для більшості (клас 0), яка залишається незмінною, оскільки це більшість, і потреби в додатковій корекції для цього класу немає;

- $class_weight_1$ — вага для меншості (клас 1), показує, що чим менше зразків класу 1 (чим більший дисбаланс), тим вищою буде вага для цього класу.

Вага для меншого класу розраховується так, щоб компенсувати різницю в кількості зразків між класами, тим самим забезпечуючи більш рівномірний вплив обох класів на побудову дерев. Це означає, що кожен об'єкт класу 1 враховується в моделі із підвищеною важливістю, що компенсує його меншу кількість у навчальних даних.

Після доповнення даних(KNN) або додавання ваг(Random Forest) обчислюються метрики точності, повноти та F1-міри для отримання збалансованої моделі.

F1-міра для кожного класу:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (12)$$

де:

- $\text{Precision} = \frac{TP}{TP+FP}$, (13)

- $\text{Recall} = \frac{TP}{TP+FN}$. (14)

Ці метрики дозволяють оцінити модель, враховуючи як позитивні, так і негативні прогнози після балансування даних.

2.7 Розрахунок середнього терміну служби на основі класифікації

Після того як моделі Random Forest та KNN навчені на збалансованих даних, проводимо розрахунок середнього терміну служби для кожного типу покриття на основі кращої моделі.

Для кращої моделі визначаємо час до зношення, при якому прогноз змінюється з 0 на 1 для кожного типу інструменту, визначається середнім значенням T_{pred} , де прогнозована ймовірність класу 1 (зношений інструмент) стає більшою або рівною 0.5.

Середній термін служби для кожного покриття можна оцінити як середнє значення для всіх об'єктів цього типу:

$$\bar{T}_{coating} = \frac{1}{N} \sum_{i=1}^N T_{pred_i} \quad (15)$$

де:

- $\bar{T}_{coating}$ — середній термін служби для конкретного покриття,
- T_{pred_i} — прогнозований термін служби для i -го об'єкта,
- N — загальна кількість об'єктів для кожного типу покриття.

Порівнявши середні значення терміну служби для кожного покриття, можна визначити, яке з них забезпечує найкращу довговічність.

Висновки до розділу 2

В даному розділі представлено підхід до прогнозування ресурсу ріжучого інструменту на основі математичного моделювання та машинного навчання. Для оцінки довговічності інструментів було використано емпіричне рівняння Тейлора, що враховує вплив швидкості різання та типу покриття на знос інструменту. Було також додано стохастичний компонент для врахування варіацій у терміні служби, що дозволяє моделі бути наближеною до реальних умов експлуатації.

На основі згенерованих синтетичних даних про знос було розроблено класифікаційні моделі з використанням методів Random Forest та K-Nearest Neighbors (KNN). Щоб підвищити точність моделей і забезпечити коректне

прогнозування для обох класів (інструмент зношений або незношений) було також здійснене балансування класів. Показана формула для розрахунку середнього терміну служби для інструментів з різними покриттями, що дозволить порівняти ефективність покриттів з точки зору довговічності.

Таким чином, розроблена модель забезпечує комплексний підхід до процесу зносу інструментів, що може слугувати основою для оптимізації процесів обробки в промислових умовах.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ

Моделювання процесу зносу ріжучих інструментів з захисними покриттями за допомогою методів машинного навчання було виконано в середовищі Python. Процес реалізації складається з декількох етапів, включаючи генерацію даних, побудову моделей класифікації на збалансованих даних, оцінку точності моделей класифікації та аналіз середньої тривалості служби на основі кращої моделі для пластини з сталі ISO P20 без покриття, з одношаровим покриттям TiN, з двошаровим покриттям TiN/ α -Al₂O₃ і з тришаровим покриттям TiN/ α -Al₂O₃ / TiCN. Для розрахунків використовувалися наведені в розділі 2 математичні співвідношення. Код програми представлений в додатку А.

Враховуючи, що різні типи пластин відрізняються лише параметрами зносу та кривою зносу інструменту, а розподіл згенерованих даних залишається тим самим для всіх типів пластин, додаткові графіки не є необхідними. Точки даних моделюють випадкову природу зносу на основі швидкості різання та часу вимірювання, що уніфіковано для всіх типів інструментів. Тому, достатньо побудувати графіки лише для сталі ISO P20, адже різниця між типами пластин відображена у різних параметрах зносу (наприклад, параметри n та C у Табл. 3.1), які враховані в кривих зносу. Такий підхід спрощує аналіз та зосереджує увагу на класифікаційних моделях, оскільки дані зберігають той самий розподіл, що й необхідно для ефективного навчання моделей.

3.1 Генерація даних про знос інструменту



Тип покриття 	ISO P20	ISO P20/ TiN	ISO P20/ TiN/ α -Al ₂ O ₃	ISO P20/ TiN/ α - Al ₂ O ₃ /TiCN
Параметри 				
n	0.4	0.41	0.43	0.45
C	183	198	229	244

Табл.3.1. Параметри n та C для різних типів різця

На рисунку 3.1 показана середня справжня довговічність інструменту як функція швидкості різання. Справжня крива ресурсу інструменту створена, щоб перевірити точність моделей машинного навчання за відомим результатом. Невизначеність була прийнята рівною 20%, щоб отримати дані про відмову та відсутність відмови там де точки не є ідеально роздільними. Це допомагає продемонструвати ефективність моделей машинного навчання в умовах невизначеності, коли дані не є лінійно роздільними.

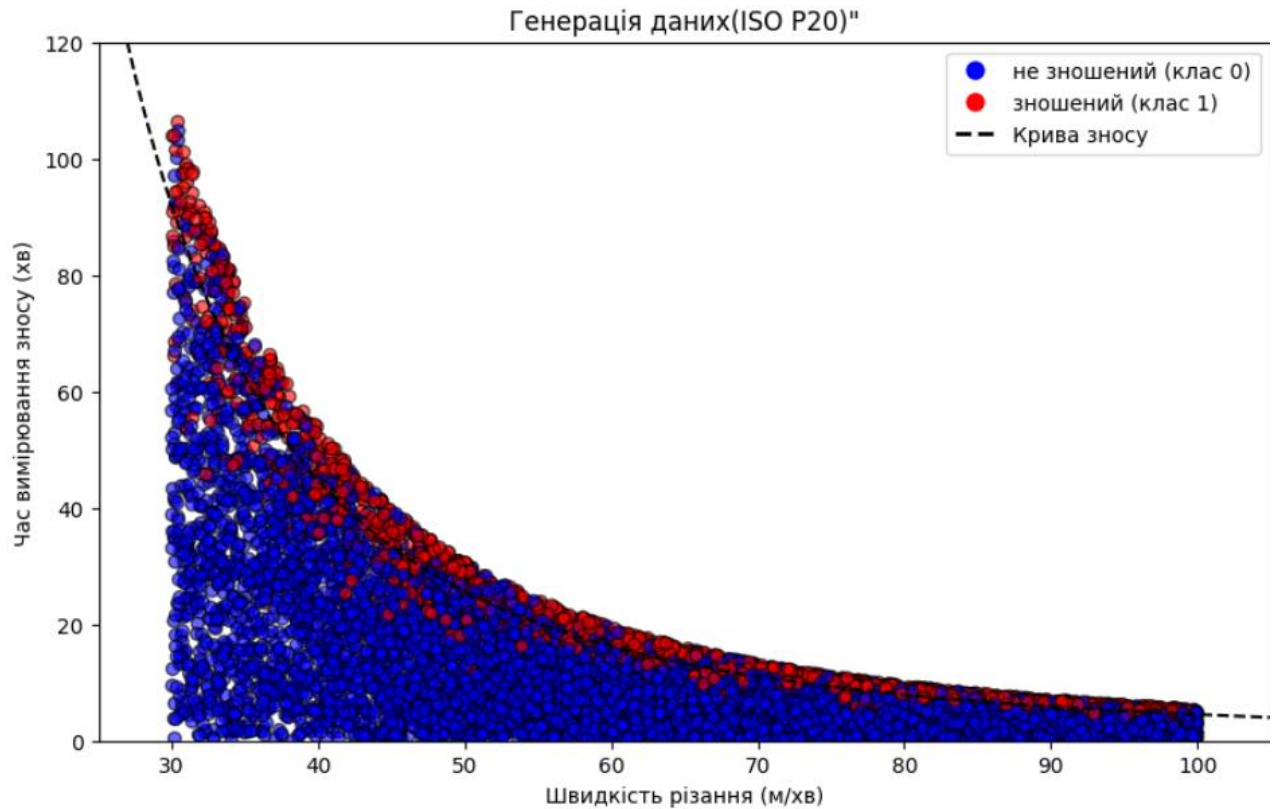


Рис.3.1 Крива зносу інструменту як функція швидкості різання і 10 000 згенерованих точок даних(ISO P20)

```
DataFrame(ISO P20)
  Швидкість різання (м/хв)  Реальний час різання  Термін служби \
0      71.52                9.29                14.16
1      37.20               10.55               90.57
2      41.74               43.47               32.78
3      63.39                7.68               19.40
4      53.84               26.71               29.69

  Стан інструменту  Клас
0      Not Failed   0
1      Not Failed   0
2      Failed       1
3      Not Failed   0
4      Not Failed   0
```

Табл.3.2.Перші 5 значень з DataFrame(ISO P20)

Набір даних ділиться на два класи: клас 0 для інструментів, що не зносились, і клас 1 для зношених інструментів. Це визначається за умовою, коли реальний час різання(t) < терміну служби(T_{sample}) для класу 0 і реальний час різання(t) > терміну служби(T_{sample}) для класу 1.(табл. 3.2.)

3.2. Методи машинного навчання для класифікації зносу інструменту

Кожен запис зносу інструмента є єдиною точкою на графіку в залежності від часу різання і дає інформацію про те, чи інструмент вийшов з ладу, чи ні. Тому термін служби інструмента розглядається як задача класифікації. Для розв'язку даної задачі обираються два методи класифікації машинного навчання: метод випадкових лісів і метод найближчих сусідів.

3.2.1 Метод випадкових лісів(Random Forest)

Реалізований метод випадкових лісів для класифікації стану інструменту на основі швидкості різання та часу вимірювання зношення, використовуючи бібліотеку `scikit-learn` для навчання моделі на згенерованих даних, що відображають стохастичну природу зносу інструментів(рис. 3.3).Кількість дерев-200,глибина дерева-5,кількість листків-2. Для балансування класів налаштовуємо ваги за допомогою параметра `class_weight`, що дозволяє збільшити вагу менших класів під час навчання, компенсуючи дисбаланс без створення додаткових даних. Після додавання ваг обчислюються метрики точності, повноти та F1-міри для отримання збалансованої моделі.

Рис.3.3 Класифікація методом випадкових лісів:

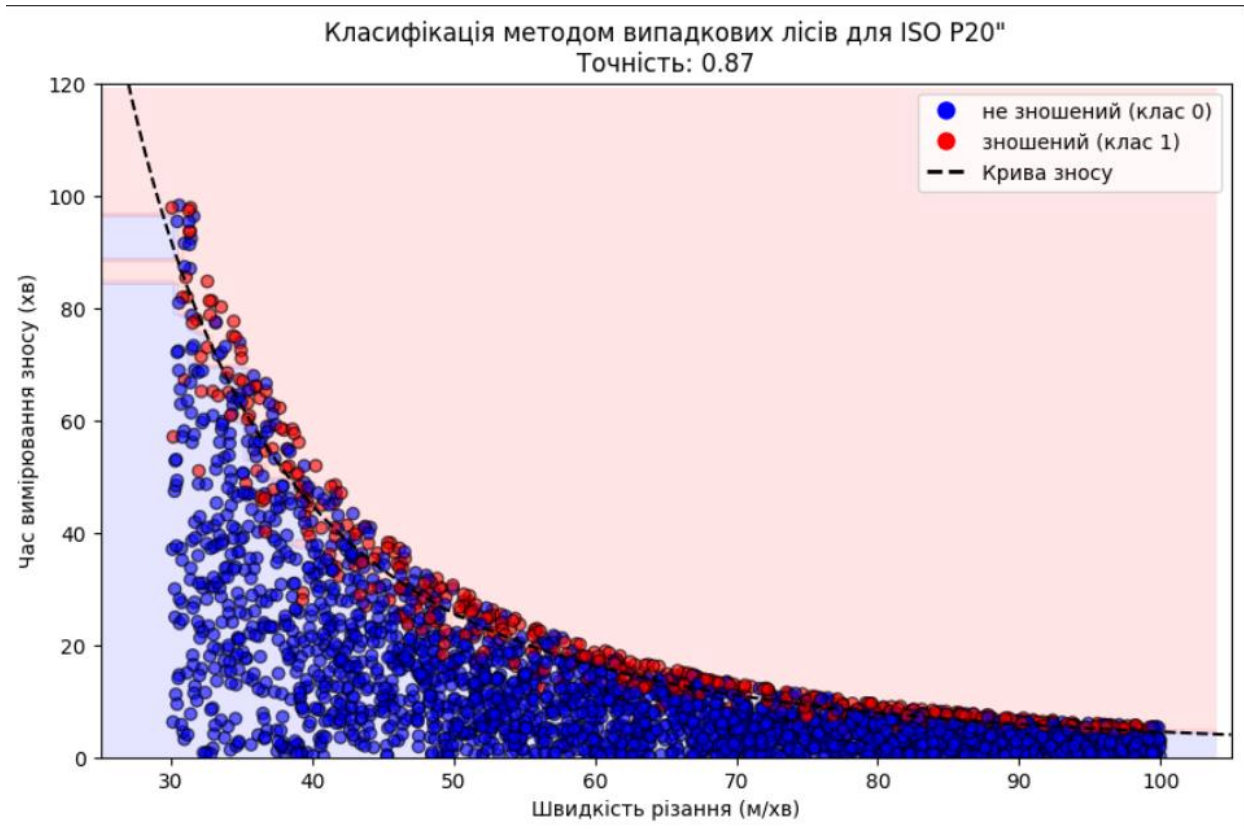


Рис. 3.4 Метрики точності, повноти та F1-міри(Random Forest)

```
Accuracy of Random Forest: 0.87
Classification Report:

```

	precision	recall	f1-score	support
0	0.93	0.91	0.92	2457
1	0.63	0.69	0.66	543
accuracy			0.87	3000

3.2.2 Метод найближчих сусідів(KNN)

Реалізований метод найближчих сусідів для класифікації стану інструменту на основі швидкості різання та часу вимірювання зношення, використовуючи бібліотеку scikit-learn для навчання моделі на згенерованих даних, що відображають стохастичну природу зносу інструментів (рис. 3.5). Кількість сусідів – 3. Для балансування класів використовується метод SMOTE, який дозволяє збільшити кількість представників меншого класу за допомогою

синтетичних даних, зберігаючи структуру простору ознак. Після застосування SMOTE обчислюються метрики точності, повноти та F1-міри для отримання збалансованої моделі.

Рис.3.5 Класифікація методом найближчих сусідів:

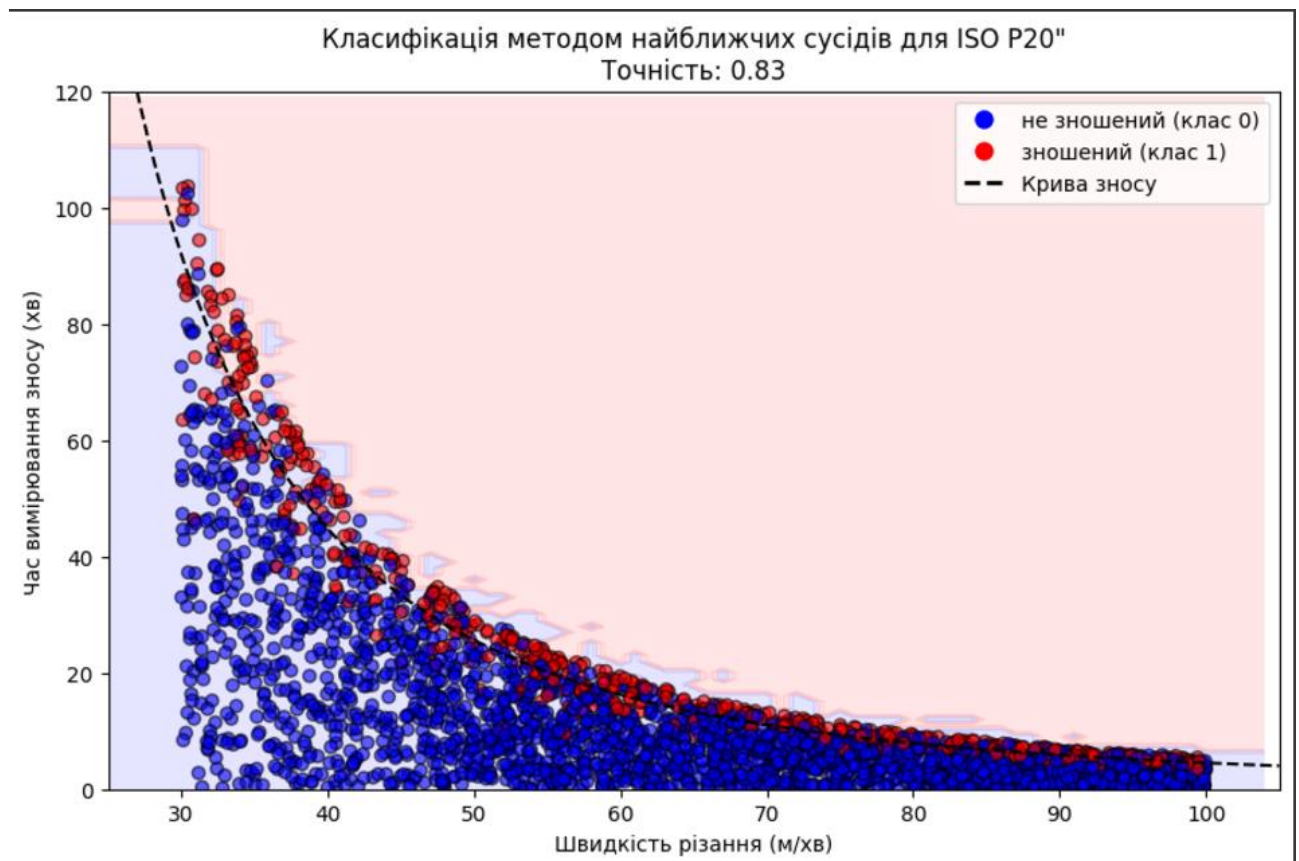


Рис. 3.6 Метрики точності, повноти та F1-міри(KNN)

```
Accuracy of KNN: 0.83
Classification Report:
```

	precision	recall	f1-score	support
0	0.94	0.84	0.89	2443
1	0.53	0.78	0.63	557
accuracy			0.83	3000
macro avg	0.74	0.81	0.76	3000
weighted avg	0.87	0.83	0.84	3000

3.3 Аналіз середньої тривалості служби на основі кращої моделі

Для аналізу середньої тривалості служби інструменту використано метод випадкових лісів, оскільки його точність виявилась вищою (0,87) порівняно з методом найближчих сусідів (0,83). Прогнозування ймовірностей зношення для тестової вибірки здійснюється для класу зношеного інструменту, після чого за умовою ймовірності зношення $\geq 0,5$ прогнозується термін служби інструменту на основі часу вимірювання зношення. Середній термін служби для інструменту кожного типу розраховується як середнє значення прогнозованих термінів служби для тих інструментів, що вважаються зношеними.

Отримані дані(рис. 3.7-3.8) дозволяють оцінити вплив покриття на тривалість служби інструменту: зі збільшенням кількості шарів покриття спостерігається збільшення середнього терміну служби інструменту.

Середній термін служби для 'ISO P20': 23.52 хвилин

Середній термін служби для 'ISO P20/ TiN': 25.94 хвилин

Середній термін служби для 'ISO P20/ TiN/ α -Al2O3': 31.87 хвилин

Середній термін служби для 'ISO P20/ TiN/ α -Al2O3/TiCN': 32.04 хвилин

Рис. 3.7 Результати розрахунків середнього терміну служби для різних типів інструментів

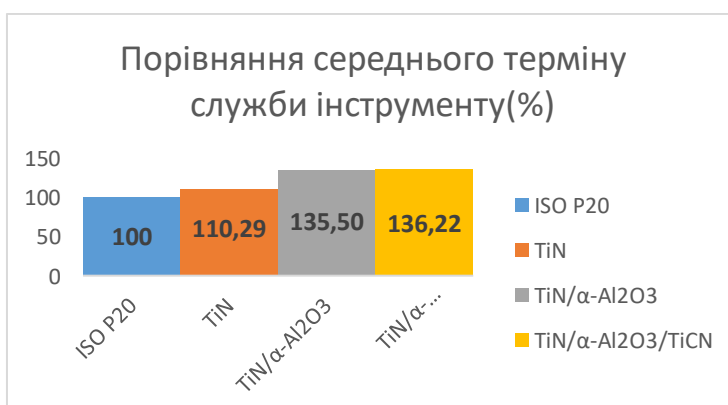


Рис. 3.8 Порівняння середнього терміну служби інструменту в залежності від покриття

ВИСНОВКИ

Під час переддипломної практики було проведено моделювання процесу зносу ріжучого інструменту без захисного покриття, з одношаровим та багатошаровими покриттями.

Для математичного моделювання зношування ріжучих інструментів були використані два методи класифікації—метод випадкових лісів і метод найближчих сусідів. Проведений аналіз показав, що метод випадкових лісів демонструє вищу точність класифікації зношеного та незношеного інструменту, що може бути використано для більш ефективного моніторингу стану інструментів під час обробки.

Встановлено, що багатошарові покриття покращують середній термін служби ріжучих інструментів. Це підтверджує доцільність використання складних покриттів для зниження зносу та збільшення продуктивності.

Для подальшого підвищення точності прогнозів терміну служби інструментів доцільно розробляти гібридні моделі, які можуть поєднувати декілька підходів машинного навчання та більш точно враховувати зміни експлуатаційних умов.

СПИСОК ВИКРИСТАНИХ ДЖЕРЕЛ

1. Tlusty J. Manufacturing Process and Equipment, Prentice Hall, Upper saddle River, NJ: pp. 463, 2000
2. Taylor FW. On the Art of Cutting Metals, Transactions of the ASME, 1906; 28:31-248.
3. Özel T, Karpat Y. Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks. International Journal of Machine Tools and Manufacture. 2005; 1;45(4-5):467-79.
4. Quiza R, Figueira L, Davim JP. Comparing statistical models and artificial neural networks on predicting the tool wear in hard machining D2 AISI steel. The International Journal of Advanced Manufacturing Technology. 2008; 1;37(7-8):641-8.
5. Alauddin M, El Baradie MA, Hashmi MS. Prediction of tool life in end milling by response surface methodology. Journal of Materials Processing Technology. 1997;71(3):456-65.
6. Ginta TL, Amin AK, Radzi HC, Lajis MA. Tool life prediction by response surface methodology in end milling titanium alloy Ti-6Al-4V using uncoated WC-Co inserts. European Journal of Scientific Research. 2009;28(4):533-41.
7. Usui E, Shirakashi T, Kitagawa T. Analytical prediction of cutting tool wear. Wear. 1984;;100(1-3):129-51.
8. Usui E, Shirakashi T, Kitagawa T. Analytical prediction of three dimensional cutting process—Part 3: Cutting temperature and crater wear of carbide tool. Journal of Engineering for industry. 1978;100(2):236-43.
9. Xie LJ, Schmidt J, Schmidt C, Biesinger F. 2D FEM estimate of tool wear in turning operation. Wear. 2005;;258(10):1479-90.

10. Quinto DT. Mechanical property and structure relationships in hard coatings for cutting tools. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*. 1988 May;6(3):2149-57.
11. Karandikar JM, Abbas A, Schmitz TL. Remaining useful tool life predictions in turning using Bayesian inference. *International Journal of Prognostics and Health Management*. 2012; 025:2153-2648.
12. Karandikar J (2019) Machine learning classification for tool life modeling using production shop-floor tool wear data. *Procedia Manuf* 34:446–454.
<https://doi.org/10.1016/j.promfg.2019.06.192>
13. Samuel AL. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*. 1959;3:3:210-29.
14. Mitchell TM. *Machine learning*. 1997. Burr Ridge, IL: McGraw Hill. 1997;45:37:870-877
15. SY. Liang, RL. Hecker, RG. Landers “Machining process monitoring and control: the state-of-the-art,” *Trans ASME Journal of Manufacturing Science Engineering*, 126–2, 2004, pp. 297–310.
16. AG. Ulsoy, Y. Koren “Control of machining processes,” *ASME Journal Dynamic System Measurement Control*, 115, 1993, 301–8.
17. Altintas Y. “Manufacturing automation: metal cutting mechanics,” machine tool vibrations, and CNC design, Cambridge University Press; 2000, pp 100-103.
18. A. Matsubara, S. Ibaraki “Monitoring and control of cutting forces in machining processes: a review,” *International Journal of Automation Technology* 2009, pp 445–456.
19. A. Matsubara, “Current status and trends of monitoring and control technology in machining process,” *Journal Society Instrument Control Engineering*, Vol 41 2002, pp 781–786.

20. E. Dimla, S. Dimla, "Sensor signals for tool-wear monitoring in metal cutting operations—a review of methods," *International Journal of Machine Tools & Manufacture*, Vol 40, 2000, pp 1073–1098.
21. P.W. Prickett, C. Johns, "An overview of approaches to end milling tool monitoring," *International Journal of Machine Tools & Manufacture*, Vol 39, 1999, pp 105–122.
22. S. Ibaraki, T. Shimizu, "A long-term control scheme of cutting forces to regulate tool life in end milling processes," *Precision Engineering*, Vol 34 2010, pp 675–682.
23. W. Yan, Y.S. Wong, K.S. Lee, T. Ning, "An investigation of indices based on milling force for tool wear in milling," *Journal of Materials Processing Technology*, Vol 89, 1999, pp, 245–253.
24. P. Bhattacharyya, D. Sengupta, S. Mukhopadhyay, "Cutting forcebased real-time estimation of tool wear in face milling using a combination of signal processing techniques," *Mechanical Systems and Signal Processing*, Vol 21, 2007, pp 2665–2683.
25. M. Kiouss, A. Ouahabi, M. Boudraa, R. Serra, A. Cheknane, "Detection process approach of tool wear in high speed milling," *Measurement*, Vol 43, 2010, pp 1439–1446.
26. N. Ghosh, Y.B. Ravi, A. Patra, S. Mukhopadhyay, S. Paul, A.R. Mohanty, A.B. Chattopadhyaya, "Estimation of tool wear during CNC milling using neural network-based sensor fusion," *Mechanical Systems and Signal Processing*, Vol 21, 2007, pp 466–479.
27. D.E. Dimla Sr. a, P.M. Lister, "On-line metal cutting tool condition monitoring.II: tool-state classification using multi-layer perceptron neural networks," *International Journal of Machine Tools & Manufacture*, Vol 40, 2000, pp 769–781.

28. M. Balazinski, E. Czogala, K. Jemielniak, J. Leski, "Tool condition monitoring using artificial intelligence methods," *Engineering Applications of Artificial Intelligence*, Vol 15 , 2002, pp 73–80.
29. T. J. Ko, D. W. Cho, "Estimation of tool wear length in finish milling using a fuzzy inference algorithm," *Wear*, Vol 169, 1993, pp 97- 106.
30. R. E. Haber, A. Alique, "Intelligent process supervision for predicting tool wear in machining processes," *Mechatronics*, Vol 13, 2003, pp 825–849.
31. L. Ning, St. C. Veldhuis, "Mechanistic Modeling of Ball End Milling Including Tool Wear," *Journal of Manufacturing Processes*, Vol. 8, 2006, pp 21-28.
32. T. Ozel, A. Nadgir, "Prediction of flank wear by using back propagation neural network modeling when cutting hardened H-13 steel with chamfered and honed CBN tools," *International Journal of Machine Tools & Manufacture*, Vol 42, 2002, pp 287–297.
33. U. Zuperl, F. Cus, "Tool cutting force modeling in ball-end milling using multilevel perceptron," *Journal of Materials Processing Technology*, Vol 153–154, 2004, pp 268–275.
34. A. Rivero, L.N. López de Lacalle, M. L. Penalv, "Tool wear detection in dry high-speed milling based upon the analysis of machine internal signals," *Mechatronics*, Vol 18, 2008, pp 627–633.
35. S. Orhan, A. Osman Er, N. Camus-cu, E. Aslan, "Tool wear evaluation by vibration analysis during end milling of AISI D3 cold work tool steel with 35 HRC hardness," *NDT&E International*, Vol 40, 2007, pp 121–126.
36. P. Wilkinson, R. L. Reuben, "Tool wear perdition from acoustic emission and surface characteristic via an artificial neural network," *Mechanical Systems and Signal Processing*, 13(6), 1999, pp 955-966.

37. Devillez, A., & Rancourt, C. (2013). "Modeling of Tool Wear in Machining." *The International Journal of Advanced Manufacturing Technology*, 64(9-12), 1501-1512.
38. Karpuschewski, B., & Nitz, J. (2014). "Influence of Coatings on the Wear Resistance of Cutting Tools." *Journal of Materials Processing Technology*, 214(5), 1058-1067.
39. Yilmaz, T., & Gok, M. (2018). "The Effects of Coatings on the Performance of Cutting Tools." *Materials Science and Engineering: A*, 718, 67-76.
40. Bäcker, M., & Bäcker, M. (2016). "Wear Behavior of Cutting Tools with Multilayer Coatings." *Wear*, 366-367, 130-139.
41. Breiman, L. (2001). "Random Forests." *Machine Learning*, 45(1), 5-32. doi: 10.1023/A:1010933404324.
42. Cover, T., & Hart, P. (1967). "Nearest Neighbor Pattern Classification." *IEEE Transactions on Information Theory*, 13(1), 21-27. doi: 10.1109/TIT.1967.1053964.

ДОДАТОК А

```

# Генерація даних
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
from matplotlib.colors import ListedColormap
from sklearn.neighbors import KNeighborsClassifier
from imblearn.over_sampling import SMOTE

# Параметри для "Carbide ISO P20"
n = 0.4
C = 183
num_samples = 10000 # Кількість згенерованих точок даних

# Функція для обчислення ресурсу інструменту з урахуванням стохастичної
природи
def tool_life(v, n, C):
    mean_T = (C / v) ** (1 / n)
    stddev_T = 0.2 * mean_T # Стандартне відхилення - 20% від середнього
значення
    return mean_T, stddev_T

# Генерація випадкових точок даних
cutting_speeds_random = np.random.uniform(30, 100, num_samples)
tool_life_samples = []
wear_measurement_times = []
tool_status = []

for v in cutting_speeds_random:
    mean_T, stddev_T = tool_life(v, n, C)
    tool_life_samples.append(np.random.normal(mean_T, stddev_T)) # Зразок
часу життя інструменту
    wear_measurement_times.append(np.random.uniform(0, 1.2) * mean_T) #
Випадковий час вимірювання зношення

# Порівняння часу життя і часу вимірювання
tool_status = [1 if measurement_time > life_time else 0 for
measurement_time, life_time in zip(wear_measurement_times,
tool_life_samples)]

# Створення DataFrame з даних
data = {
    'Швидкість різання (м/хв)': cutting_speeds_random,
    'Реальний час різання': wear_measurement_times,

```



```

    'Термін служби': tool_life_samples,
    'Стан інструменту': tool_status
}
df = pd.DataFrame(data)

# Розділення на навчальну і тестову вибірки
X = df[['Швидкість різання (м/хв)', 'Реальний час різання']]
y = df['Стан інструменту']

# Візуалізація розділяючої кривої
x_min, x_max = 25, 105 # Розширено межі осі X
y_min, y_max = 0, 120 # Встановлено межі осі Y
xx, yy = np.meshgrid(np.arange(x_min, x_max, 1), np.arange(y_min, y_max, 1))
Z = rf_model.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)

# Візуалізація розділяючої кривої з правильними кольорами
plt.figure(figsize=(10, 6))

# Відображення точок даних
plt.scatter(X['Швидкість різання (м/хв)'], X['Реальний час різання'], s=y,
            cmap=ListedColormap(['blue', 'red']), edgecolor='k', alpha=0.6)

# Додавання кривої зносу, що охоплює весь графік
cutting_speeds_curve = np.linspace(x_min, x_max, 500) # Розширено до x_min
та x_max
mean_tool_life_curve = [(C / v) ** (1 / n) for v in cutting_speeds_curve]
plt.plot(cutting_speeds_curve, mean_tool_life_curve, color='black',
         linestyle='--', label='Крива зносу')

# Налаштування меж графіка
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)

plt.xlabel('Швидкість різання (м/хв)')
plt.ylabel('Час вимірювання зносу (хв)')
plt.title('Генерація даних (ISO P20)')
plt.legend(handles=[
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='blue',
              markersize=10, label='не зношений (клас 0)'),
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='red',
              markersize=10, label='зношений (клас 1)'),
    plt.Line2D([0], [0], color='black', linestyle='--', linewidth=2,
              label='Крива зносу')
], loc='upper right')
plt.show()

```

```

# Округлення чисел до 2 знаків після коми
df = df.round(2)

# Додаємо колонку 'Class' на основі колонки 'Tool Status'
df['Стан інструменту'] = df['Стан інструменту'].map({0: 'Not Failed', 1:
'Failed'})
df['Клас'] = np.where(df['Стан інструменту'] == 'Failed', 1, 0)

# Виведення перших рядків DataFrame для кожного випадку
print("DataFrame (ISO P20)")
print(df.head())

```

```

# Навчання моделі Random Forest

# Параметри для "Carbide ISO P20"
n = 0.45
C = 244
num_samples = 10000 # Кількість згенерованих точок даних

# Функція для обчислення ресурсу інструменту з урахуванням стохастичної
природи
def tool_life(v, n, C):
    mean_T = (C / v) ** (1 / n)
    stddev_T = 0.2 * mean_T # Стандартне відхилення - 20% від середнього
значення
    return mean_T, stddev_T

# Генерація випадкових точок даних
cutting_speeds_random = np.random.uniform(30, 100, num_samples)
tool_life_samples = []
wear_measurement_times = []
tool_status = []

for v in cutting_speeds_random:
    mean_T, stddev_T = tool_life(v, n, C)
    tool_life_samples.append(np.random.normal(mean_T, stddev_T)) # Зразок
часу життя інструменту
    wear_measurement_times.append(np.random.uniform(0, 1.2) * mean_T) #
Випадковий час вимірювання зношення

# Порівняння часу життя і часу вимірювання
tool_status = [1 if measurement_time > life_time else 0 for
measurement_time, life_time in zip(wear_measurement_times,
tool_life_samples)]

# Створення DataFrame з даних
data = {
    'Cutting Speed (m/min)': cutting_speeds_random,

```

```

    'Wear Measurement Time': wear_measurement_times,
    'Tool Life Sample': tool_life_samples,
    'Tool Status': tool_status
}
df = pd.DataFrame(data)

# Розділення на навчальну і тестову вибірки
X = df[['Cutting Speed (m/min)', 'Wear Measurement Time']]
y = df['Tool Status']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

# Навчання моделі Random Forest з урахуванням зважування класів
class_weights = {0: 1, 1: len(y) / (2 * np.bincount(y)[1])} # Зважування
класів
rf_model = RandomForestClassifier(n_estimators=200,
class_weight=class_weights, min_samples_split=5, min_samples_leaf=2,
random_state=42)
rf_model.fit(X_train, y_train)

# Прогноз на тестовій вибірці
y_pred = rf_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)

print(f"Accuracy of Random Forest: {accuracy:.2f}")
print("Classification Report:\n", classification_report(y_test, y_pred))

# Візуалізація розділяючої кривої
x_min, x_max = 25, 105 # Розширено межі осі X
y_min, y_max = 0, 120 # Встановлено межі осі Y
xx, yy = np.meshgrid(np.arange(x_min, x_max, 1), np.arange(y_min, y_max, 1))
Z = rf_model.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)

# Візуалізація розділяючої кривої з правильними кольорами
plt.figure(figsize=(10, 6))

# Виправлення кольорової палітри: синя область для класу Not Failed, червона
для Failed
cmap_light = ListedColormap(['#AAAAFF', '#FFAAAA']) # Світло-блакитний для
Not Failed, світло-рожевий для Failed
plt.contourf(xx, yy, Z, cmap=cmap_light, alpha=0.3)

# Відображення точок даних
plt.scatter(X_test['Cutting Speed (m/min)'], X_test['Wear Measurement
Time'], c=y_test, cmap=ListedColormap(['blue', 'red']), edgecolor='k',
alpha=0.6)

# Додавання кривої зносу, що охоплює весь графік

```

```

cutting_speeds_curve = np.linspace(x_min, x_max, 500) # Розширено до x_min
та x_max
mean_tool_life_curve = [(C / v) ** (1 / n) for v in cutting_speeds_curve]
plt.plot(cutting_speeds_curve, mean_tool_life_curve, color='black',
linestyle='--', label='Крива зносу')

# Налаштування меж графіка
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)

plt.xlabel('Швидкість різання (м/хв)')
plt.ylabel('Час вимірювання зносу (хв)')
plt.title(f'Класифікація методом випадкових лісів для ISO P20"\nТочність:
{accuracy:.2f}')
plt.legend(handles=[
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='blue',
markersize=10, label='не зношений (клас 0)'),
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='red',
markersize=10, label='зношений (клас 1)'),
    plt.Line2D([0], [0], color='black', linestyle='--', linewidth=2,
label='Крива зносу')
], loc='upper right')
plt.show()

# Розрахунок середнього терміну служби(Random Forest)
# Прогнозування ймовірностей зношення для тестової вибірки
probabilities = rf_model.predict_proba(X_test)[: , 1] # Отримуємо
ймовірності для класу 1 (зношений інструмент)

# Знаходимо прогнозовані терміни служби, коли ймовірність зношення ≥ 0.5
predicted_life = []
for i, prob in enumerate(probabilities):
    if prob >= 0.5:
        predicted_life.append(X_test['Wear Measurement Time'].iloc[i]) #
Використовуємо час вимірювання зношення

# Розрахунок середнього терміну служби для зношених інструментів
mean_tool_life = np.mean(predicted_life) if predicted_life else 0

print(f"Середній термін служби для 'ISO P20/ TiN/α-Al2O3/TiCN':
{mean_tool_life:.2f} хвилин")

```

```

# Навчання моделі K-Nearest Neighbors

# Параметри для "Carbide ISO P20"
n = 0.4
C = 183
num_samples = 10000 # Кількість згенерованих точок даних

```

```

# Функція для обчислення ресурсу інструменту з урахуванням стохастичної
природи
def tool_life(v, n, C):
    mean_T = (C / v) ** (1 / n)
    stddev_T = 0.2 * mean_T # Стандартне відхилення - 20% від середнього
значення
    return mean_T, stddev_T

# Генерація випадкових точок даних
cutting_speeds_random = np.random.uniform(30, 100, num_samples)
tool_life_samples = []
wear_measurement_times = []
tool_status = []

for v in cutting_speeds_random:
    mean_T, stddev_T = tool_life(v, n, C)
    tool_life_samples.append(np.random.normal(mean_T, stddev_T)) # Зразок
часу життя інструменту
    wear_measurement_times.append(np.random.uniform(0, 1.2) * mean_T) #
Випадковий час вимірювання зношення

# Порівняння часу життя і часу вимірювання
tool_status = [1 if measurement_time > life_time else 0 for
measurement_time, life_time in zip(wear_measurement_times,
tool_life_samples)]

# Створення DataFrame з даних
data = {
    'Cutting Speed (m/min)': cutting_speeds_random,
    'Wear Measurement Time': wear_measurement_times,
    'Tool Life Sample': tool_life_samples,
    'Tool Status': tool_status
}
df = pd.DataFrame(data)

# Розділення на навчальну і тестову вибірки
X = df[['Cutting Speed (m/min)', 'Wear Measurement Time']]
y = df['Tool Status']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

# Застосування SMOTE для балансування класів
smote = SMOTE(random_state=42)
X_train_balanced, y_train_balanced = smote.fit_resample(X_train, y_train)

# Навчання моделі K-Nearest Neighbors
knn_model = KNeighborsClassifier(n_neighbors=3)
knn_model.fit(X_train_balanced, y_train_balanced)

# Прогноз на тестовій вибірці

```

```

y_pred = knn_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)

print(f"Accuracy of K-Nearest Neighbors: {accuracy:.2f}")
print("Classification Report:\n", classification_report(y_test, y_pred))

# Візуалізація розділяючої кривої
x_min, x_max = 25, 105 # Розширено межі осі X
y_min, y_max = 0, 120 # Встановлено межі осі Y
xx, yy = np.meshgrid(np.arange(x_min, x_max, 1), np.arange(y_min, y_max, 1))
Z = knn_model.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)

# Візуалізація розділяючої кривої з правильними кольорами
plt.figure(figsize=(10, 6))
cmap_light = ListedColormap(['#AAAAFF', '#FFAAAA']) # Світло-блакитний для
Not Failed, світло-рожевий для Failed
plt.contourf(xx, yy, Z, cmap=cmap_light, alpha=0.3)

# Відображення точок даних
plt.scatter(X_test['Cutting Speed (m/min)'], X_test['Wear Measurement
Time'], c=y_test, cmap=ListedColormap(['blue', 'red']), edgecolor='k',
alpha=0.6)

# Додавання кривої зносу, що охоплює весь графік
cutting_speeds_curve = np.linspace(x_min, x_max, 500) # Розширено до x_min
та x_max
mean_tool_life_curve = [(C / v) ** (1 / n) for v in cutting_speeds_curve]
plt.plot(cutting_speeds_curve, mean_tool_life_curve, color='black',
linestyle='--', label='Крива зносу')

# Налаштування меж графіка
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)

plt.xlabel('Швидкість різання (м/хв)')
plt.ylabel('Час вимірювання зносу (хв)')
plt.title(f'Класифікація методом найближчих сусідів для ISO P20\nТочність:
{accuracy:.2f}')
plt.legend(handles=[
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='blue',
markersize=10, label='не зношений (клас 0)'),
    plt.Line2D([0], [0], marker='o', color='w', markerfacecolor='red',
markersize=10, label='зношений (клас 1)'),
    plt.Line2D([0], [0], color='black', linestyle='--', linewidth=2,
label='Крива зносу')
], loc='upper right')
plt.show()

# Розрахунок середнього терміну служби(KNN)
# Прогнозування ймовірностей зношення для тестової вибірки

```

```
probabilities = rf_model.predict_proba(X_test)[: , 1] # Отримуємо
ймовірності для класу 1 (зношений інструмент)

# Знаходимо прогнозовані терміни служби, коли ймовірність зношення ≥ 0.5
predicted_life = []
for i, prob in enumerate(probabilities):
    if prob >= 0.5:
        predicted_life.append(X_test['Wear Measurement Time'].iloc[i]) #
Використовуємо час вимірювання зношення

# Розрахунок середнього терміну служби для зношених інструментів
mean_tool_life = np.mean(predicted_life) if predicted_life else 0

print(f"Середній термін служби для 'Carbide ISO P20': {mean_tool_life:.2f}
хвилин")
```