# ДОСЛІДЖЕННЯ МЕТОДІВ ЗМЕНШЕННЯ НАВЧАЛЬНОЇ ВИБІРКИ ДАНИХ

**УДК 004.9**

**DOI: https://doi.org/10.35546/2313-0687.2020.27.98-107**

**Яків Повод,**
аспірант кафедри програмних засобів і технологій, ХНТУ, Херсон, Україна,
**E-mail:** elornau@gmail.com, ORCID ID: 0000-0001-8403-5945

**Володимир Шерстюк,**
доктор технічних наук, професор кафедри програмних засобів і технологій, ХНТУ, Херсон, Україна,
**E-mail:** vgsherstyuk@gmail.com, ORCID ID: 0000-0002-9096-2582

**Анотація.** Значна кількість методів машинного навчання мають обмеження на об'єм даних з якими вони можуть працювати. Зазвичай ці обмеження проявляють себе як надмірне споживання розрахункових ресурсів, або пам'яті. Так, як більшість алгоритмів машинного навчання мають розрахункову складність більшу ніж O(n), при значному об'єму вхідних даних, ці алгоритми не зможуть знайти рішення за розумний час. Зменшення навчальної вибірки для цих алгоритмів підвищить швидкість їх роботі пропорційно до розрахункової складності алгоритмів.

У статі проаналізовано методи зменшення навчальної вибірки для деяких алгоритмів машинного навчання. Виміряно вплив зменшення навчальної вибірки на швидкодію та точність алгоритмів машинного навчання.

Метою даного дослідження є дослідження впливу різних алгоритмів зменшення начальної вибірки на взаємну точність різних моделей машинного навчання при прогнозуванні сонячної інсоляції.

Основні результати дослідження. Досліджено вплив начальної вибірки при наявності надлишкового об'єму даних на швидкодію алгоритмів машинного навчання, та на їх точність. Виміряно вплив прокляття розмірeності при використанні значно зменшеної навчальної вибірки.

Науковою новизною є порівняння методів зменшення навчальної вибірки для передбачення сонячної інсоляції.

***Ключові слова:*** *машинне навчання, швидкодія, дані, сонячна інсоляція.*

**Постановка проблеми.** Одним із методів передбачення сонячної інсоляції є використання машинного навчання на попередньо виміряних статистичних даних.

Існує значна кількість методів машинного навчання. Більшість цих методів можуть дати непогані результати, проте для різних наборів даних точність та швидкість роботи різних алгоритмів машинного навчання можуть відрізнятися. Додатково точність та швидкодія можуть відрізнятися від параметрів алгоритмів машинного навчання. Так лише вибірка даних з 219 метеостанцій дає майже 2 мільйона точок. Використання більшої бази даних може призвести до того, що знайти оптимальну модель та параметри цієї моделі методом перебору буде неможливо за раціональний період часу.

Отже додатково було вирішено перевірити як змінюється взаємна точність моделей при зменшенні розміру навчальної вибірки.

**Аналіз останніх досліджень і публікацій.** Хоча з проблемою надмірного обсягу даних стикається велика кількість досліджень у різних сферах, на даний момент не має універсального метода вирішення цієї проблеми.

Питання зменшення навчальної вибірки даних піднімається в роботах Патель Ф.Н., Барбара Д., Дюмушель В., Фалуцос К., Хаас П., Хеллерстайн Дж., Іоаннідіс Ю., Севчик К. Серед останніх публікацій, в [9] було описано метод зменшення вибірки даних LiDAR, та порівняно з методом випадкової вибірки.

Зазвичай використання всього об'єму доступних даних дає найкращі результати. Проте в ситуаціях коли використовувати повний набір даних є неможливим. Одним з методів вирішення цього питання – є використання випадкової вибірки даних. Ефективність цього метода для задач кластеризації було досліджено в [6]. Цей метод дає можливість ігнорувати надлишкову інформацію, яка є у вхідних даних.

Хоча ігнорування надлишкової інформації дає змогу підвищити швидкодію моделі, цей метод може призвести до втрати важливих ознак, так як всі точки мають рівноцінне значення. Проте Хокон Кіле та Кєтіл Улен [5] запропонували використовувати кластеризацію для збереження важливих характерних елементів з набору даних. Цей метод припускає, що вхідні дані мають у своєму складі певну кількість шуму, і для фільтрації цього шуму використовуються центри кластерів.

**Мета дослідження.** Зменшення навчальної вибірки зменшує об'єм інформації для навчання моделі машинного навчання, що має знизити її точність Отже метою дослідження являється вимір впливу зменшення об'єму навчальної вибірки на точність роботи моделей для прогнозування сонячної інсоляції. Також дослідження впливу зменшення навчальної вибірки на взаємну точність моделей машинного навчання. А також дослідження можливості збільшення швидкості знаходження оптимальних гіперпараметрів за допомогою зменшення навчальної вибірки.

**Виклад матеріалу дослідження.** В якості вхідних даних були обрані статистичні данні для прогнозування сонячної інсоляції. Ці данні отримані від найточніших метеостанцій з відібраних типових метеорологічних даних. Всього в наборі 1918438 точок. При чому кожна точка має 6 вхідних даних а саме: кут нахилу сонця над лінією горизонту, площа непрозорого хмарного покрову, загальна площа хмарного покрову, висота над рівнем моря, азимут на сонце, сонячне випромінювання без врахування атмосфери та 2 вихідних: Пряме сонячне випромінювання (DNI) та розсіяне сонячне випромінювання (DHI).

Для перевірки моделей дані було розділено на тестову та тренувальну групи. Розмір тестової групи складає 20% від тренувальної. Різні алгоритми були порівняні за наступними критеріями: пояснена дисперсія (EV), середнє квадратичне відхилення (RMSE), медіанна абсолютна похибка (MAE), коефіцієнт детермінації (R2), час навчання (TL) та час тестування на 525.6 тисячах точок (TT). Результати розрахунків занесені в таблицю 1.

Для більшої наочності для кожної моделі були побудовані графіки залежності вихідних даних від куту нахилу сонця над лінією горизонту, як найбільш впливової змінної (рис. 1–5).

Як можна зазначити з графіків, не зважаючи на високі тестові результати алгоритми на базі дерев рішень страждають від шуму, а також потребують значно більше розрахункових потужностей для передбачення значень у порівнянні із багатошаровим перцептором. Також найточніші алгоритми потребують значних ресурсів при навчанні. Додатково підвищити якість моделей можна за допомогою підбору гіперпараметрів,

**Таблиця 1** – Метрики моделей побудованих на базі повного набору даних

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.625287 | 122.878952 | 0.625284 | 0.3283 | 0.0150 |
| Поліноміальна регресія | 0.816719 | 85.742865 | 0.816717 | 2.0109 | 0.2062 |
| Регресія деревом рішень | 0.814194 | 83.357597 | 0.814191 | 16.4844 | 0.0490 |
| Random Forest | 0.903148 | 60.150910 | 0.903146 | 1069.5313 | 3.5013 |
| Багатошаровий персептрон | 0.896980 | 62.005843 | 0.896858 | 1030.8302 | 0.5856 |

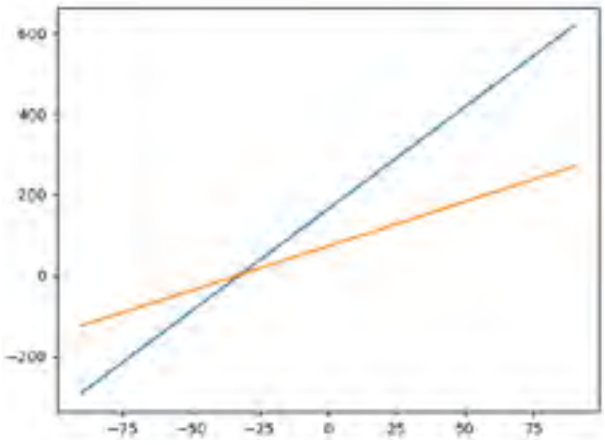

*Рисунок 1 – Лінійна регресія*

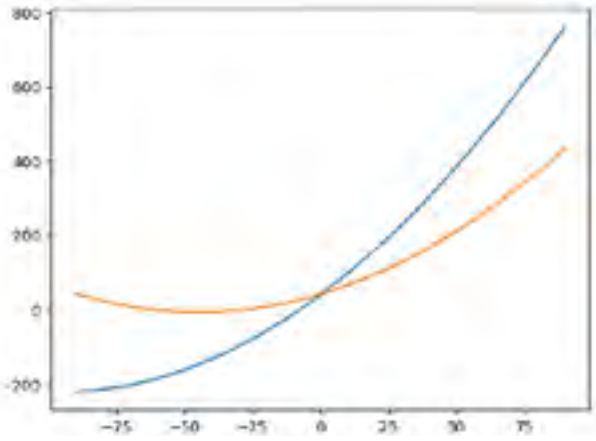

*Рисунок 2 – Поліноміальна регресія*

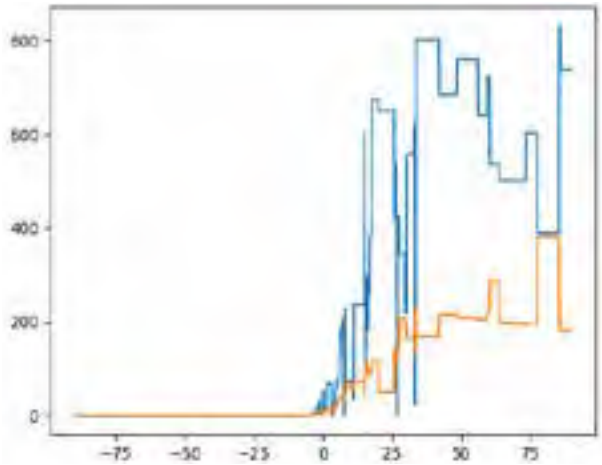

*Рисунок 3 – Регресія деревом рішень*

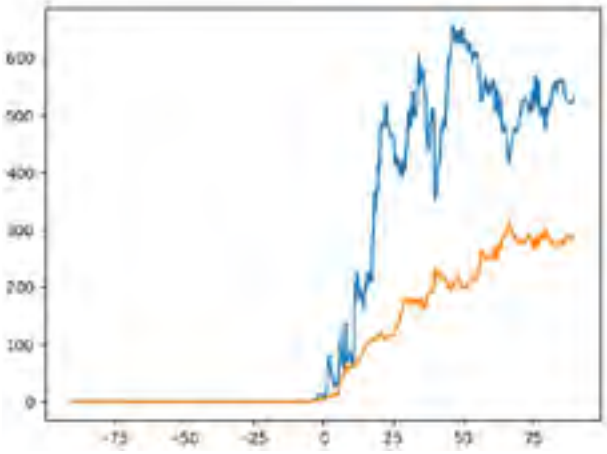

*Рисунок 4 – Random Forest*

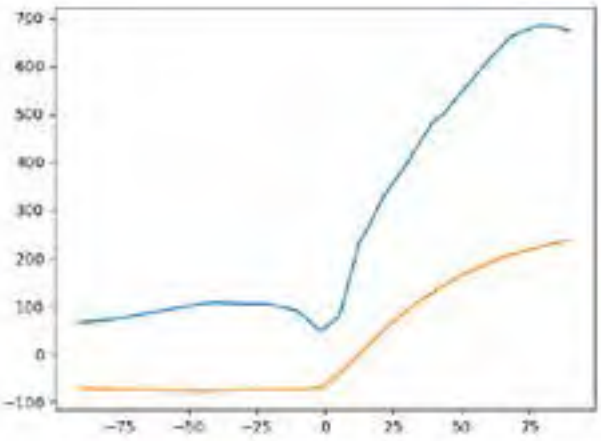

*Рисунок 5 – Регресія за допомогою багатошарового персептрона*

проте із за довгого навчання на це необхідно витратити значну кількість розрахункових ресурсів.

Для вирішення цієї проблеми було запропоновано навчати модель лише на невеликій частині випадково обраних даних.

Так як точок тепер значно менше, процес навчання моделі проводитися із значно більшою швидкістю. Це дозволяє спробувати підбір гіперпараметрів для оптимізації цих алгоритмів. Отже були побудовані моделі для наборів даних з 46656, 3125 та 256 точками.

Результати перевірки цих моделей на такому самому наборі тестових даних, який проводився при перевірці моделей побудованих на повному наборі даних занесено в таблиці 2–4.

**Таблиця 2** – Метрики моделей побудованих на базі випадкової вибірки даних для 46656 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.625251 | 122.882795 | 0.625250 | 0.0380 | 0.0130 |
| Поліноміальна регресія | 0.816603 | 85.773208 | 0.816601 | 0.0470 | 0.1782 |
| Регресія деревом рішень | 0.791224 | 88.413800 | 0.791216 | 0.2993 | 0.0210 |
| Random Forest | 0.890927 | 63.873009 | 0.890925 | 16.1743 | 2.4668 |
| Багатошаровий персептрон | 0.889021 | 64.410215 | 0.888368 | 13.4437 | 0.5400 |

**Таблиця 3** – Метрики моделей побудованих на базі випадкової вибірки даних для 3125 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.624931 | 122.965483 | 0.624764 | 0.0010 | 0.0120 |
| Поліноміальна регресія | 0.815037 | 86.184080 | 0.815024 | 0.0050 | 0.1797 |
| Регресія деревом рішень | 0.766261 | 94.170411 | 0.766249 | 0.0120 | 0.0210 |
| Random Forest | 0.882222 | 66.752122 | 0.882217 | 0.7667 | 1.6160 |
| Багатошаровий персептрон | 0.811762 | 86.029162 | 0.811069 | 1.8928 | 0.5170 |
| Random Forest 350 | 0.883062 | 66.502719 | 0.883054 | 1607.0191 | 5.9240 |
| Оптимізований персептрон | 0.737982 | 93.107640 | 0.737975 | 630.7818 | 0.0641 |

**Таблиця 4** – Метрики моделей побудованих на базі випадкової вибірки даних для 256 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.614858 | 124.644501 | 0.614039 | 0.0010 | 0.0130 |
| Поліноміальна регресія | 0.773191 | 94.145401 | 0.772790 | 0.0010 | 0.1732 |
| Регресія деревом рішень | 0.695505 | 104.227557 | 0.695310 | 0.0020 | 0.0160 |
| Random Forest | 0.853006 | 73.328920 | 0.852704 | 0.1391 | 1.0500 |
| Багатошаровий персептрон | 0.568294 | 132.175444 | 0.567450 | 0.2082 | 0.5455 |
| Random Forest 60 | 0.852155 | 73.460619 | 0.851886 | 345.2720 | 0.6236 |
| Оптимізований персептрон | 0.614437 | 124.702865 | 0.613774 | 146.2547 | 0.0561 |

Для більшої наочності були побудовані графіки для моделей Random Forest та багатошарового персептрона для 46656 та 256 точок (рис. 6–9).

Використання випадкової вибірки дозволяє значно підвищити швидкість навчання, при цьому це не має значного впливу на точність. Так алгоритм Random Fores навчений на 256 точках дав результат значно кращій ніж лінійна регресія, знайдена за повним набором даних, при цьому час навчання моделі на випадковій вибірці був навіть менший. Оптимізація гіперпараметрів для алгоритму Random Fores в даному випадку не дала значного впливу на точність, а при оптимізації гіперпараметрів багатошарового персептрона більший вплив мало випадкове початкове значення при ініціалізації вагів синапсів.

Так як данні зашумлені, збільшення точності при збільшенні розміру тренувальної вибірки може зумовлюватися тим, що модель отримує більше інформації для зменшення шуму. Розділення даних на кластери за допомогою методу к–середніх, та використання центральних точок кластерів має згладити шуми та значно зменшити кількість точок. Проте використання звичайного метода к–середніх, для такого об'єму даних призводить до необхідності використовувати

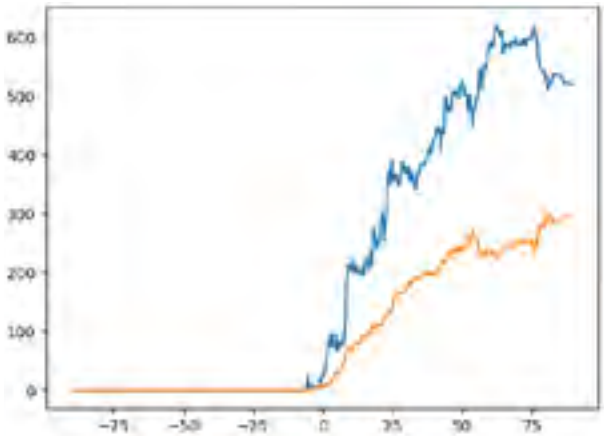

*Рисунок 6 – Регресія Random Forest за 46656 точками*

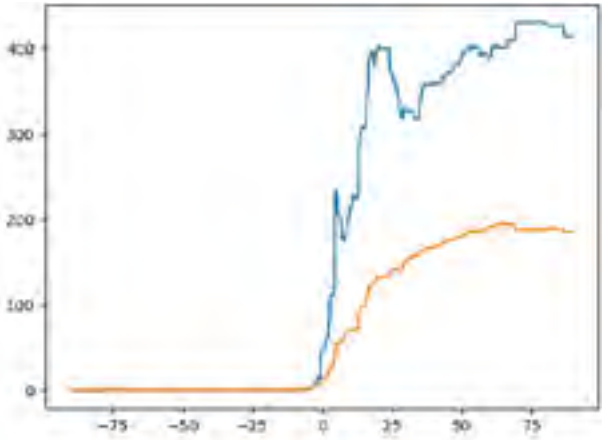

*Рисунок 7 – Регресія Random Forest за 256 точками*

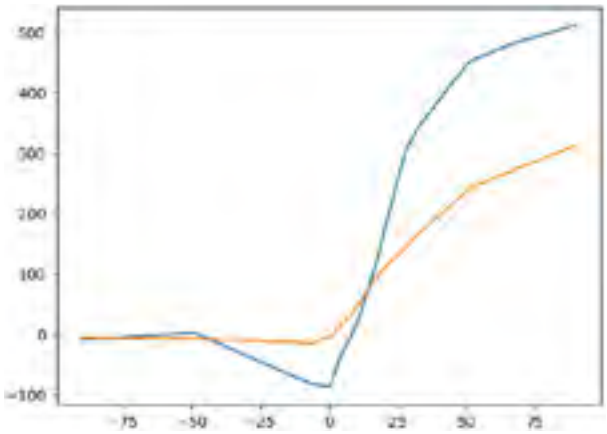

*Рисунок 8 – Багатошаровий персептрон за 46656 точками*

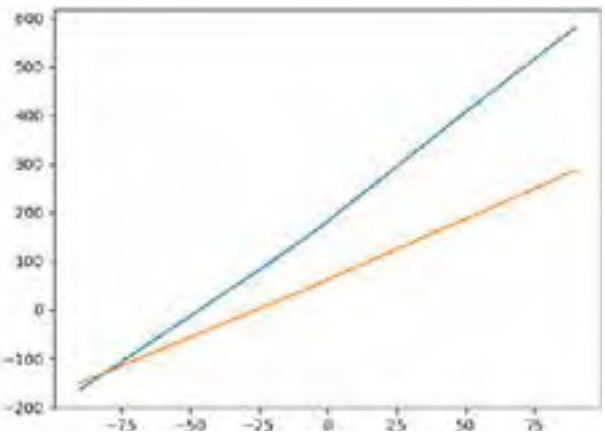

*Рисунок 9 – Багатошаровий персептрон за 256 точками*

надмірно багато розрахункових ресурсів. Також розділення даних на пакети, та послідовне уточнення за допомогою методу к–середніх центрів кластерів дає значне підвищення швидкодії, причому має не значно зменшувати точність результату. Отже дані були розділені на 46656, 3125, 256 точок. Час, який було затрачено на кластеризацію в залежності від кількості точок занесено в таблицю 5.

**Таблиця 5** – Залежність часу кластеризації від кількості вихідних точок

| Кількість точок | Час, с. |
|---|---|
| 46656 | 3302.5406 |
| 3125 | 36.1358 |
| 256 | 68.4481 |
| 27 | 4.5222 |

Відповідно до цієї таблиці можна зробити висновок, що існує нелінійна залежність від кількості кластерів. Це може впливати на те, що існує верхня границя кількості кластерів, на яку вже не доцільно розбивати модель, так як проведення всіх необхідних досліджень можна буде провести на повному наборі даних за менший час.

Результати моделей побудованих за допомогою вибірки даних кластеризацією занесені до таблиць 6-8.

Для більшої наочності були побудовані графіки для моделей Random Forest та багатошарового персептрона для 46656 та 256 точок (рис. 10–13).

**Таблиця 6** – Метрики моделей побудованих на базі кластеризованої вибірки даних для 46656 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.491590 | 144.681186 | 0.490259 | 0.0090 | 0.0120 |
| Поліноміальна регресія | 0.791288 | 91.276680 | 0.790489 | 0.0450 | 0.1772 |
| Регресія деревом рішень | 0.794314 | 87.620376 | 0.794089 | 0.3889 | 0.0340 |
| Random Forest | 0.888659 | 64.565105 | 0.888445 | 22.2032 | 2.5003 |
| Оптимізований персептрон | 0.885234 | 65.467897 | 0.885113 | 15.3079 | 0.5405 |

**Таблиця 7** – Метрики моделей побудованих на базі кластеризованої вибірки даних для 3125 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | 0.456674 | 149.370304 | 0.455312 | 0.0010 | 0.0120 |
| Поліноміальна регресія | 0.776504 | 93.418590 | 0.775220 | 0.0050 | 0.1812 |
| Регресія деревом рішень | 0.784864 | 89.245941 | 0.784185 | 0.0170 | 0.0190 |
| Random Forest | 0.878035 | 67.582027 | 0.877275 | 1.0660 | 1.4609 |
| Багатошаровий персептрон | 0.725737 | 107.539230 | 0.724772 | 1.8152 | 0.5045 |
| Random Forest 140 | 0.878700 | 67.388178 | 0.877965 | 2167.8188 | 1.9868 |
| Оптимізований персептрон | 0.661927 | 119.171927 | 0.658966 | 723.3256 | 0.0600 |

**Таблиця 8** – Метрики моделей побудованих на базі кластеризованої вибірки даних для 256 точок

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Лінійна регресія | -0.027458 | 206.751648 | -0.027687 | 0.0000 | 0.0120 |
| Поліноміальна регресія | -0.716148 | 241.150022 | -0.845401 | 0.0020 | 0.1837 |
| Регресія деревом рішень | 0.661489 | 127.412505 | 0.610520 | 0.0150 | 0.0140 |
| Random Forest | 0.633758 | 137.734278 | 0.566412 | 0.1551 | 0.9694 |
| Багатошаровий персептрон | 0.567952 | 131.486897 | 0.565914 | 0.2092 | 0.5330 |
| Random Forest 330 | 0.635827 | 137.401464 | 0.568320 | 375.0439 | 3.2040 |
| Оптимізований персептрон | 0.193946 | 188.250720 | 0.193721 | 176.8661 | 0.0571 |

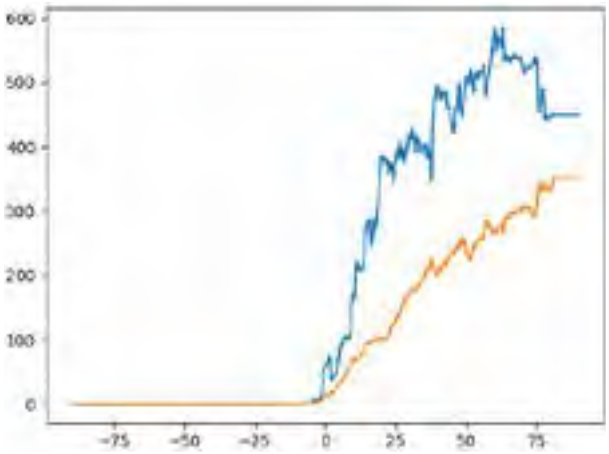


*Рисунок 10 – Регресія Random Forest за 46656 точками*

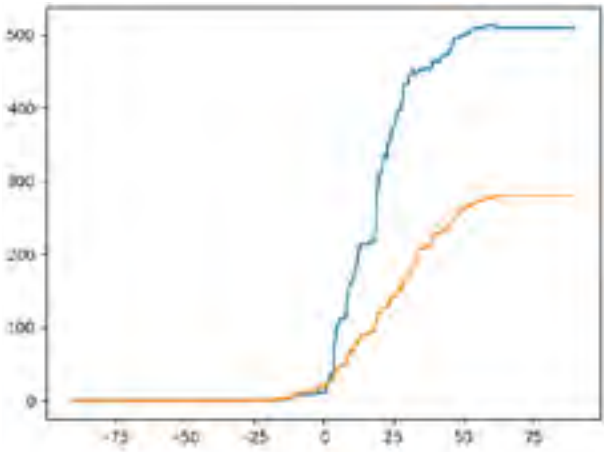


*Рисунок 11 – Регресія Random Forest за 256 точками*

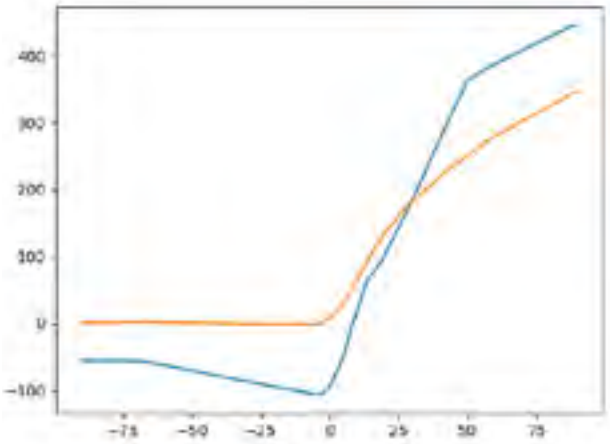

*Рисунок 12 – Багатошаровий персептрон за 46656 точками*

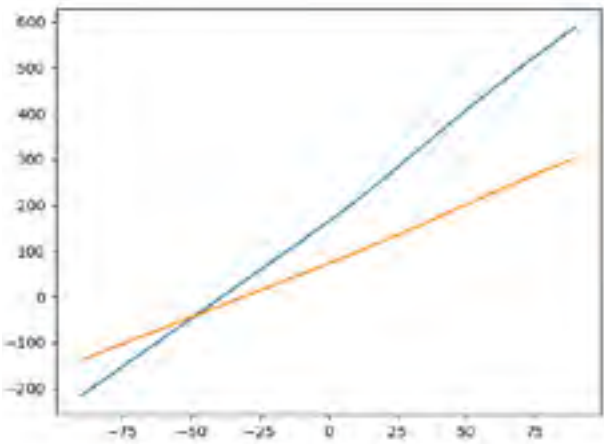

*Рисунок 13 – Багатошаровий персептрон за 256 точками*

Не зважаючи на те, що кластеризовані дані мають бути менш зашумленими, точність одержаних моделей знаходиться на рівні близькому до даних отриманих випадковою вибіркою, а в деяких випадках навіть у гіршому стані.

Причиною цього може бути те, що данні отримані за допомогою кластеризації мають розподіл, який відрізняється від розподілу оригінального набору даних.

Також пресептрон значно втрачає точність при зменшенні розміру вибірки. Причиною цього може бути «прокляття розмірності», для перевірки цього, можна зменшити розмірність вхідних даних. Для зменшення розмірності використовувався алгоритм для знаходження оптимальної комбінації вхідних параметрів заданої кількості. Залежність коефіцієнта детермінації від кількості вимірів та кількості точок зображено в таблиці 9 та 10.

**Таблиця 9** – Точність багатошарового персептрона на випадковій вибірці даних

| | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|
| 46656 | 0.889021 | 0.887369 | 0.882440 | 0.886156 | 0.883539 | 0.642373 |
| 3125 | 0.811762 | 0.815706 | 0.834550 | 0.828208 | 0.834565 | 0.630194 |
| 256 | 0.568048 | 0.523668 | 0.594506 | 0.640980 | 0.635581 | 0.612262 |
| 27 | 0.067735 | 0.018385 | 0.021285 | 0.243897 | 0.543966 | 0.518505 |
| 4 | -2.751225 | -1.518638 | -2.472216 | -11.229447 | 0.540164 | 0.531449 |

**Таблиця 10** – Точність багатошарового персептрона на кластеризованих даних

| | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|
| 46656 | 0.885234 | 0.883998 | 0.884393 | 0.880177 | 0.880290 | 0.639043 |
| 3125 | 0.850628 | 0.857905 | 0.844665 | 0.837346 | 0.873672 | 0.632749 |
| 256 | 0.341854 | 0.538805 | 0.772292 | 0.795018 | 0.741028 | 0.606975 |
| 27 | 0.203687 | 0.189355 | 0.592604 | 0.663000 | 0.439170 | 0.532177 |
| 4 | 0.009273 | -0.146042 | 0.201778 | 0.389213 | 0.303835 | -0.277406 |

Згідно з даних таблиць 9 та 10 – точність моделей на невеликій вибірці даних при зменшенні розмірності вхідних даних збільшується. Що можна інтерпретувати як те, що моделі на базі багатошарового персептрона страждають від «прокляття розмірності».

Окрім вибору моделі, важливим фактором можуть бути також гіперпараметри цієї моделі. Хоча оптимізація гіперпараметрів для моделі Random Forest не дала надійного результату, для багатошарового персептрону більшість вибірок показувала найкращі резуль-

тати на однакових параметрах. Отже було вирішено перевірити ці параметри при навчанні моделі на повному наборі даних та побудовано графік оптимізованої моделі.

**Таблиця 11** – Порівняння регресії багатошаровим персептроном з оптимізованими гіперараметрами

| Модель | EV | RMSE | R2 | TL | TT |
|---|---|---|---|---|---|
| Не оптимізована модель | 0.896980 | 62.005843 | 0.896858 | 1030.8302 | 0.5856 |
| Оптимізована модель | 0.898714 | 61.485744 | 0.898712 | 1908.8021 | 1.6761 |

Також для перевірки було побудовано графік залежності куту сонця від кількості енергії для багатошарового персептрона.

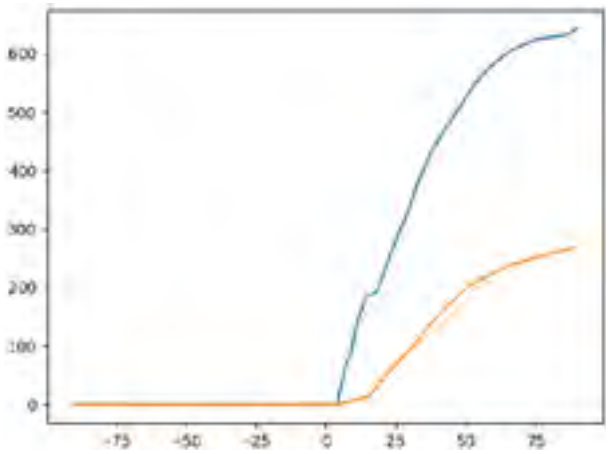


***Рисунок 14 – Багатошаровий персептрон з оптимізованими гіперпараметрами***

Використання оптимізованих гіперпараметрів дещо підвищило точність моделі, а також модель з оптимізованими гіперпараметрами має найбільш близький вигляд до того, як має виглядати ідеальна модель.

**Висновки.**

1. Зменшення розміру навчальної вибірки дало змогу перевірити значну кількість моделей а також оптимізувати гіперпараметри цих моделей. Що дало змогу отримати більш точну модель, а також графік оптимізованої моделі має найбільшу схожість до ідеального.

2. Якщо є можливість пожертвувати незначними втратами в точності, то можна скористатися випадковою вибіркою даних для навчання моделі. Це дозволить в десятки або навіть в сотні разів зменшити час навчання.

3. Якщо об'єм даних на стільки великий, що використання складних моделей для навчання на цьому наборі є неможливим, використання складних моделей натренованих на випадковій вибірці даних може дати кращій результат ніж навчання більш простих моделей на повному наборі даних

4. Мінімальний розмір вибірки обмежено розмірністю вхідних даних та мінімальною кількістю точок, яка необхідна для роботи алгоритму.

5. Використання кластеризації для зменшення вибірки даних для цього набору даних дає результати кращі за випадкову вибірку лише для багатошарового персептрона, та при умові зменшення розмірності вхідних даних

6. Оптимізація гіперпараметрів та вибір моделі можна виконувати за допомогою зменшеної вибірки даних, так як точність різних моделей зберігається на широкому діапазоні розмірів вхідних даних. Проте, при значному зменшенні розміру вхідних даних, або при великій кількості вимірів вхідних даних на точність починає впливати «прокляття розмірності».

## СПИСОК ЛІТЕРАТУРИ

1. Barbara D. та ін. The new Jersey data Reduction Report [Електронний ресурс]. URL: https://www.aminer.cn/pub/53e9a6aeb7602d9702fe32b6/the-new-jersey-data-reduction-report (дата звернення: 29.04.2021).
2. Błaszczak-Bąk W. та ін. Down-sampling of large lidar dataset in the context of off-road objects extraction // Geosciences. 2020. Т. 10. № 6. С. 219.
3. Cebeci Z., Yildiz F. Efficiency of random sampling based data size reduction on computing time and validity of clustering in data mining // Journal of Agricultural Informatics. 2016. Т. 7. № 1.
4. Harrell F.E. Regression modeling Strategies // Springer Series in Statistics. 2001.
5. Ingrassia S., Morlini I. Neural network modeling for small datasets // Technometrics. 2005. Т. 47. № 3. С. 297–311.

6. Kile H., Uhlen K. Data reduction via clustering and averaging for contingency and reliability analysis // International Journal of Electrical Power & Energy Systems. 2012. Т. 43. № 1. С. 1435–1442.
7. Patel F.N. Large high dimensional data handling using data reduction // 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT). 2016.
8. Pestov V. Is thek-nn classifier in high dimensions affected by the curse of dimensionality? // Computers & Mathematics with Applications. 2013. Т. 65. № 10. С. 1427–1437.
9. Sculley D. Web-scale k-means clustering // Proceedings of the 19th international conference on World wide web – WWW '10. 2010.

## RESEARCH OF METHODS OF REDUCTION OF EDUCATIONAL SAMPLING OF DATA

**Yakiv Povod,**
Postgraduate Student of the Department of Software and Technologies, KNTU,
Kherson, Ukraine,
e-mail: elornau@gmail.com, ORCID ID: 0000-0001-8403-5945

**Volodymyr Sherstyuk,**
Doctor of Technical Sciences, Professor of the Department of Software and Technologies, KNTU, Kherson, Ukraine,
e-mail: vgsherstyuk@gmail.com, ORCID ID: 0000-0002-9096-2582

**Abstract.** Many machine learning methods have limitations on the amount of data they can work with. These limitations usually appear as excessive consumption of computing resources or memory. Since most machine learning algorithms have a computational complexity greater than O (n), with a significant amount of input data, these algorithms will not be able to find a solution in a reasonable amount of time. Reducing the training sample for these algorithms will increase the computational speed in proportion to the calculated complexity of the algorithms.

The article analyzes the methods of reducing the sample for some machine learning algorithms. The influence of training sample reduction on the speed and accuracy of machine learning algorithms was measured.

The purpose of this study is to investigate the effect of different algorithms for reducing the initial sample on the mutual accuracy of different models of machine learning models for predicting solar insolation.

The main results of the study. The influence of the initial sample in the presence of excess data on the speed of machine learning algorithms and their accuracy has been studied. The effect of the dimensional curse was measured using a significantly reduced training sample.

A scientific novelty is the comparison of methods for reducing the educational sample to predict solar insolation.

***Key words:*** *machine learning, computational speed, data sampling, solar insolation.*

## ИССЛЕДОВАНИЕ МЕТОДОВ СНИЖЕНИЯ ОБУЧАЮЩЕЙ ВЫБОРКИ ДАННЫХ

**Яков Повод,**
аспирант кафедры программных средств и технологий, ХНТУ, Херсон, Украина,
e-mail: elornau@gmail.com, ORCID ID: 0000-0001-8403-5945

**Владимир Шерстюк,**
доктор технических наук, профессор кафедры программных средств и технологий, ХНТУ, Херсон, Украина,
e-mail: vgsherstyuk@gmail.com, ORCID ID: 0000-0002-9096-2582

**Аннотация.** Значительное количество методов машинного обучения имеют ограничения на объем данных с которыми они могут работать. Обычно эти ограничения проявляют себя как чрезмерное потребление вычислительных ресурсов, или памяти. Так, как большинство алгоритмов машинного обучения имеют вычислительную

сложность больше чем O (n), при значительном объема входных данных, эти алгоритмы не смогут найти решение за разумное время. Уменьшение обучающей выборки для этих алгоритмов повысит скорость их работы пропорционально расчетной сложности алгоритмов.

В статье проанализированы методы уменьшения обучающей выборки для некоторых алгоритмов машинного обучения. Измерено влияние уменьшения обучающей выборки на быстродействие и точность алгоритмов машинного обучения.

Целью данного исследования является исследование влияния различных алгоритмов уменьшения начального выборки на взаимную точность различных моделей машинного обучения при прогнозировании солнечной инсоляции.

Основные результаты исследования. Исследовано влияние начального выборки при наличии избыточного объема данных на быстродействие алгоритмов машинного обучения, и на их точность. Измерения влияние проклятие размеренности при использовании значительно уменьшенной обучающей выборки.

Научной новизной является сравнение методов уменьшения обучающей выборки для предсказания солнечной инсоляции.

***Ключевые слова:*** *машинное обучение, быстродействие, данные, солнечная инсоляция.*

## REFERENCES

1. Barbara, D., Dumouchel, W., Faloutsos, C., Haas, P., Hellerstein, J., Ioannidis, Y., Sevcik, K. (n.d.). The new Jersey data Reduction Report. Retrieved April 29, 2021, from https://www.aminer.cn/pub/53e9a6aeb7602d9702fe32b6/the-new-jersey-data-reduction-report
2. Błaszczak-Bąk, W., Janicka, J., Suchocki, C., Masiero, A., & Sobieraj-Żłobińska, A. (2020). Down-sampling of large lidar dataset in the context of off-road objects extraction. *Geosciences, 10(6)*, 219. doi:10.3390/geosciences10060219
3. Cebeci, Z., & Yildiz, F. (2016). Efficiency of random sampling based data size reduction on computing time and validity of clustering in data mining. *Journal of Agricultural Informatics*, 7(1). doi:10.17700/jai.2016.7.1.266
4. Harrell, F. E. (2001). Regression modeling Strategies. *Springer Series in Statistics*. doi:10.1007/978-1-4757-3462-1
5. Ingrassia, S., & Morlini, I. (2005). Neural network modeling for small datasets. *Technometrics, 47(3)*, 297–311. doi:10.1198/004017005000000058
6. Kile, H., & Uhlen, K. (2012). Data reduction via clustering and averaging for contingency and reliability analysis. *International Journal of Electrical Power & Energy Systems*, 43(1), 1435–1442. doi:10.1016/j.ijepes.2012.07.011
7. Patel, F. N. (2016). Large high dimensional data handling using data reduction. *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. doi:10.1109/iceeot.2016.7754940
8. Pestov, V. (2013). Is thek-nn classifier in high dimensions affected by the curse of dimensionality? *Computers & Mathematics with Applications, 65(10)*, 1427–1437. doi:10.1016/j.camwa.2012.09.011
9. Sculley, D. (2010). Web-scale k-means clustering. *Proceedings of the 19th International Conference on World Wide Web – WWW'10*. doi:10.1145/1772690.1772862