

В.О. Гайдукевич, А.Ю. Дорошенко

ЗАСТОСУВАННЯ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ ЕНЕРГОСПОЖИВАННЯ В СИСТЕМАХ РОЗУМНОГО ДОМУ

У статті досліджено застосування методів машинного навчання для прогнозування енергоспоживання в умовах функціонування систем розумного дому. Основу дослідження становить міжнародно відомий часовий набір даних PSML (Power System Machine Learning), який включає інформацію про споживання електроенергії, генерацію, балансування та прогнозування навантаження в умовах декарбонізованої енергетичної мережі. Набір даних PSML характеризується високою деталізацією та охоплює різні часові масштаби — від годинних до щоденних значень, що дозволяє оцінювати як короткострокові, так і середньострокові тенденції енергоспоживання.

У роботі проведено порівняльний аналіз класичних та сучасних методів машинного навчання, включно з регресією, класифікацією та кластеризацією, для прогнозування навантаження та виявлення шаблонів споживання електроенергії в побутовому середовищі. Особлива увага приділена оптимізації моделей для роботи з великими даними, обробці пропусків та аномалій, а також інтеграції прогнозів у системи автоматичного управління енергоресурсами розумного дому.

Ключові слова: машинне навчання, прогнозування, розумний дім, оптимізація енергоспоживання, PSML.

V.O. Haidukevych, A.Y. Doroshenko

APPLICATION OF MACHINE LEARNING MODELS TO PREDICT ENERGY CONSUMPTION IN SMART HOME SYSTEMS

The article investigates the application of machine learning methods to forecast energy consumption in the context of smart home systems. The research is based on the internationally renowned PSML (Power System Machine Learning) time series dataset, which includes information on electricity consumption, generation, balancing and load forecasting in the context of a decarbonized energy network. The PSML dataset is characterized by high detail and covers different time scales - from hourly to daily values, which allows assessing both short-term and medium-term trends in energy consumption.

The paper provides a comparative analysis of classical and modern machine learning methods, including regression, classification and clustering, for load forecasting and identifying patterns in electricity consumption in the domestic environment. Particular attention is paid to optimizing models for working with big data, processing gaps and anomalies, as well as integrating forecasts into automatic smart home energy management systems.

Keywords: machine learning, forecasting, smart home, energy optimization, PSML.

Вступ

Цифрова трансформація енергетичної галузі, зростання ролі децентралізованих джерел енергії та активна участь споживачів у процесах керування навантаженням актуалізують потребу у новітніх підходах до прогнозування енергоспоживання [1]. В умовах високої варіативності, сезонних коливань та метеозалежної генерації класичні статистичні методи дедалі частіше виявляються недостатньо ефективними [2]. У цьому контексті методи штучного інтелекту, зокрема, машинного навчання, відкривають нові можливості для побудови точних і гнучких моделей, здатних вияв-

ляти приховані нелінійні закономірності у часових рядах, обробляти великі обсяги даних та автоматизувати процеси прогнозування [3].

У статті представлено підхід до регресії, класифікації та кластеризації енергоспоживання на основі відкритого набору даних PSML [4], який охоплює багаторівневі часові ряди різної роздільності — від індивідуальних побутових споживачів до регіональних енергетичних систем. Особливу увагу зосереджено на моделюванні сценаріїв для розумного дому як типового елемента сучасної енергомережі. Такий

підхід дозволяє проаналізувати поведінку кінцевого споживача, протестувати ефективність алгоритмів у реалістичних умовах та оцінити їхню придатність для інтеграції в автоматизовані системи енергоменеджменту [5].

Метою дослідження є аналіз точності та генералізаційної спроможності сучасних моделей машинного навчання у вирішенні задач прогнозування навантаження, класифікації типових режимів роботи та виявлення схожих шаблонів споживання. У межах експериментальної частини було протестовано низку моделей, включаючи метод опорних векторів, алгоритми градієнтного підсилення, метод щільнісної кластеризації та інші, які оцінювалися за метриками точності, ефективності та інтерпретованості результатів [6].

Варто відзначити, що у статті [7] основний акцент зроблено на застосуванні методів машинного навчання для підвищення енергоефективності будівель шляхом оптимізації роботи інженерних систем. На відміну від цього, наше дослідження зосереджується на аналізі багаторівневих часових рядів споживання енергії, прогнозуванні навантаження та виявленні типових сценаріїв роботи побутових пристроїв у контексті розумного дому, що робить його більш орієнтованим на інтеграцію в системи енергетичного менеджменту з динамічними моделями прогнозування.

1. Системи розумного дому як контекст для прогнозування

Розумний дім — це інтегрований комплекс апаратних і програмних рішень, що забезпечують автоматизований контроль над освітленням, мікрокліматом, безпекою, мультимедійними пристроями та особливо енергоспоживанням у житловому просторі [3]. У центрі такої системи знаходиться інтелектуальний модуль керування, здатний отримувати дані з численних сенсорів, виконувати аналіз у реальному часі та ухвалювати оптимальні рішення для підвищення комфорту мешканців та ефективності використання ресурсів.

Одним із ключових напрямів розвитку розумного дому є прогнозування споживання електроенергії. Це завдання має низку стратегічно важливих переваг [8]:

- зменшення втрат енергії — точні прогнози дозволяють уникнути роботи обладнання у холостому режимі, оптимізувати цикли роботи техніки та знизити загальні витрати.
- автоматичне керування побутовими приладами — система може самостійно вмикати чи вимикати прилади, регулювати інтенсивність роботи кондиціонерів, нагрівачів та освітлення.
- адаптація до пікових навантажень у мережі — прогнозні моделі допомагають зміщувати споживання на години з меншими тарифами або меншими навантаженнями у мережі, що особливо важливо для енергетичних систем з обмеженими ресурсами.

На практиці це означає, що система розумного дому може:

- активувати обігрів або охолодження заздалегідь, з урахуванням прогнозу температури та графіку перебування мешканців;
- автоматично заряджати акумуляторні системи або електромобілі у період нічних тарифів;
- відключати непотрібні пристрої, коли мешканців немає вдома;
- інтегруватися з відновлюваними джерелами енергії, прогнозуючи потужність від сонячних панелей або вітрогенераторів.

Використання методів машинного навчання у цьому контексті дає змогу будувати адаптивні моделі, які враховують не лише часові патерни енергоспоживання, а й зовнішні фактори, зокрема, погодні умови, вологість, графік використання побутової техніки та навіть індивідуальні звички мешканців [9]. Крім того, прогнозування в розумному домі сприяє побудові мікромереж (microgrid) — локальних енергетичних систем, де баланс між генерацією та споживанням може підтримуватися автоматично. Це зменшує залежність від центральної мережі та підвищує енергетичну автономність житла [10].

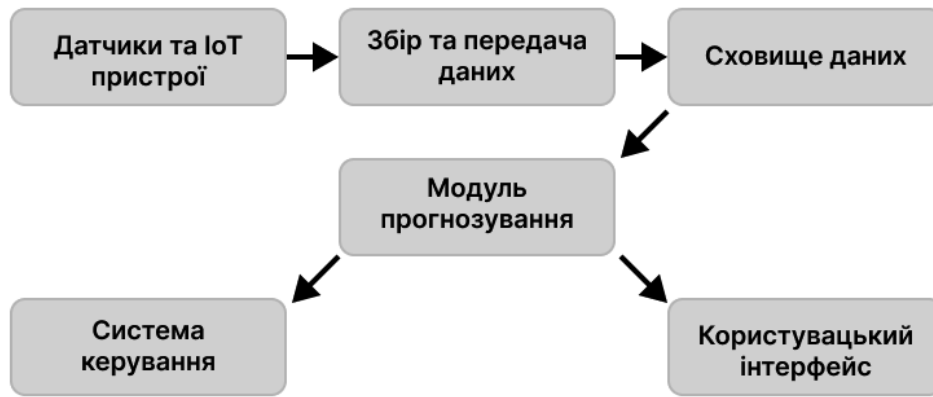


Рис. 1. Схема архітектури системи розумного дому з модулем прогнозування

2. Огляд і підготовка даних PSML

Набір PSML є спеціалізованою колекцією високодеталізованих часових рядів, які відображають динаміку побутового споживання електроенергії з інтервалом у 15 хвилин. Такі дані зазвичай надходять від сучасних розумних лічильників та систем енергомоніторингу, що інтегруються в інфраструктуру розумного дому. Цей набір містить як основні показники енергоспоживання, так і супутні фактори, що впливають на нього:

- `timestamp` — мітка часу вимірювання, яка дає змогу синхронізувати записи з іншими джерелами, наприклад прогнозами погоди або даними про генерацію електроенергії.
- `house_id` — унікальний ідентифікатор домогосподарства, що дозволяє відслідковувати індивідуальні або групові профілі.
- `power_kwh` — обсяг спожитої електроенергії (у кВт·год) за фіксований часовий інтервал; ключовий показник для побудови моделей прогнозування.
- `temperature` — температура зовнішнього середовища (°C), важлива для аналізу впливу кліматичних факторів.
- `humidity` — відносна вологість (%), параметр, що впливає на роботу кліматичних систем.
- `day_of_week`, `hour`, `is_weekend` — створені часові ознаки, що допомага-

ють враховувати добові й тижневі патерни.

Завдяки своїй структурі, PSML є типовим прикладом IoT-збірника даних, що використовується для інтелектуального управління енергоспоживанням у приватних будинках та комерційних об'єктах. Він демонструє природні коливання навантаження, зумовлені часом доби, погодними умовами та поведінкою мешканців.

Використання PSML у аналітичних платформах відкриває широкі можливості для підвищення енергоефективності. На основі цих даних можна:

- прогнозувати рівень споживання з урахуванням сезонних і погодних змін;
- автоматично регулювати роботу побутових приладів, адаптуючи їх до пікових тарифів та знижуючи витрати;
- оптимізувати роботу домашніх акумуляторних систем, визначаючи найвигідніший час для заряджання й розряджання;
- виявляти аномалії — від потенційних несправностей обладнання до нераціонального використання ресурсів.

Таким чином, цей набір даних є не лише основою для побудови прогнозних моделей, а й служить базою для створення самоадаптивних алгоритмів управління енергетикою в розумних будинках.

Для забезпечення високої якості та надійності подальшого моделювання, дані PSML пройшли ретельну попередню обробку:

1. Заповнення пропусків — відновлення відсутніх значень за допомогою інтерполяції або медіан для уникнення викривлення трендів.

2. Розширення часових ознак — з часових міток згенеровано додаткові атрибути (година доби, день тижня, вихідний день).

3. Масштабування — нормалізація числових параметрів для стабільної роботи алгоритмів машинного навчання.

4. Виділення трендів і сезонності — застосування декомпозиції часових рядів для точнішого прогнозування.

timestamp	house_id	power_kwh	temperature	humidity	day_of_week	hour	is_weekend
01-01 00:00:00	1006	1.07	24.0	59.9	0	0	False
01-01 00:15:00	1003	1.08	19.3	76.1	0	0	False
01-01 00:30:00	1007	1.66	4.1	34.4	0	0	False
01-01 00:45:00	1004	2.72	-2.1	39.8	0	0	False
01-01 01:00:00	1006	2.27	15.5	32.3	0	1	False
01-01 01:15:00	1009	1.6	8.2	46.3	0	1	False
01-01 01:30:00	1002	3.14	-1.3	49.4	0	1	False
01-01 01:45:00	1006	0.87	9.9	43.6	0	1	False
01-01 02:00:00	1007	1.6	-4.0	71.4	0	2	False
01-01 02:15:00	1004	1.96	22.3	47.8	0	2	False

Рис. 2. Фрагмент датасету PSML

3. Методика дослідження

У межах цього дослідження ставиться задача прогнозування навантаження в електричних мережах на основі розумного дому. Метою є передбачення значення енергоспоживання у майбутні часові періоди. Такий тип задач є критично важливим для планування генерації, балансування навантаження, оптимізації роботи енергетичних систем і запобігання перевантаженням.

Завданням моделі є знаходження залежності між часовими характеристиками (день тижня, година, місяць тощо) та фактичним навантаженням.

У дослідженні застосовано кілька класів алгоритмів машинного навчання для вирішення задач прогнозування, класифікації та кластеризації побутового електроспоживання на прикладі даних PSML. Підхід базується на поєднанні класичних методів обробки часових рядів та сучасних алгоритмів, здатних виявляти складні нелінійні залежності.

Для аналізу використано PSML — відкритий набір даних, що містить показники навантаження енергоспоживання з часовою роздільністю 1 година. Набір вклю-

чає дані за декілька років, що дозволяє моделі враховувати сезонні й добові коливання.

Перед моделюванням дані було нормалізовано та закодовано категоріальні змінні (наприклад, день тижня — one-hot encoding). Перед моделюванням виконано:

- нормалізацію числових ознак (Min-Max) для приведення значень у діапазон $[0,1]$ та уникнення домінування одних змінних над іншими;
- one-hot encoding для категоріальних змінних (dayofweek), щоб уникнути хибного порядкування днів;
- циклічне кодування (sin, cos) для годин та місяців, щоб зберегти їхню періодичність;
- розділення вибірки за часом (навчання на минулих даних, тестування на нових), що відповідає реальним умовам прогнозування.

Для побудови моделей машинного навчання необхідно було розділити дані на незалежні підмножини, щоб оцінка якості моделі була об'єктивною, а прогнозна здатність — прийнятною для нових даних.

У даному дослідженні застосовано наступну схему:

- навчальна вибірка — 70% даних - використовувалась для тренування моделей та підбору параметрів;
- тестова вибірка — 30% даних - призначена для остаточного тестування якості моделей на «невиданих» прикладах.

Валідаційна вибірка не виділялась окремо, оскільки її роль виконує внутрішнє розбиття у процесі крос-валідації та підбору гіперпараметрів, наприклад, із використанням алгоритму GridSearchCV.

Розбиття виконувалося за допомогою функції `train_test_split()` з бібліотеки `scikit-learn` із фіксацією параметра `random_state=42` для забезпечення відтворюваності результатів у повторних експериментах.

4. Прогнозування параметрів споживання (регресія)

Для вирішення задач прогнозування та аналізу структури даних використано наступні моделі:

1. Support Vector Regressor (SVR)

Метод опорних векторів дозволяє будувати нелінійні моделі за рахунок використання ядрових функцій. Було протестовано Gaussian RBF kernel. Модель показала позитивні результати на короткострокових прогнозах. Використані гіперпараметри: `kernel = 'rbf'` — радіальна базисна функція `C = 1000` — штраф за помилку `epsilon = 0.1` — ширина "труби" точності `gamma = 'scale'` — автоматичне визначення параметра

2. Random Forest Regressor (RFR),

Метод Випадкового Лісу як ансамблевий метод дозволив врахувати взаємозв'язки між температурою, годиною доби та обсягом споживання. RF був обраний завдяки високій стійкості до перенавчання і здатності працювати з некорельованими ознаками. Використані гіперпараметри: `n_estimators = 200` — кількість дерев `max_depth = 15` — максимальна глибина дерева `min_samples_split = 5` — мінімальна кількість прикладів для розбиття

`random_state = 42` — для стабільності результатів.

3. XGBoost Regressor

Алгоритм градієнтного підсилення над деревами рішень, який показав високу ефективність у задачах регресії на складних вибірках. Завдяки регуляризації він зменшує ризик перенавчання. Використані гіперпараметри:

`n_estimators = 300` — кількість ітерацій
`learning_rate = 0.05` — швидкість навчання
`max_depth = 10` — максимальна глибина дерев
`subsample = 0.8` — частка даних для побудови кожного дерева
`colsample_bytree = 0.8` — частка ознак для кожного дерева

У додатку 1 наведено фрагмент програмного коду на Python, який реалізує прогнозування енергоспоживання на основі метеорологічних ознак з використанням моделі Random Forest.

Для оцінки точності прогнозування використано три основні метрики регресії:

- MAE (Mean Absolute Error) — середня абсолютна похибка;
- RMSE (Root Mean Squared Error) — корінь середньоквадратичної помилки;
- R² (коефіцієнт детермінації) — частка поясненої дисперсії;
- MAPE (Mean Absolute Percentage Error) — середня абсолютна відносна похибка, яка показує похибку у відсотках від фактичного значення.

Метрики обчислювались як для тестової, так і для крос-валідаційної вибірки.

Моделі SVR, RFR та XGBoost було протестовано на одній і тій самій вибірці. Результати представлено у таблиці 1:

Таблиця 1.

Похибка регресії у прогнозуванні спожитої електроенергії (параметр `power_kwh`)

Модель	MAE (кВт)	RMSE (кВт)	R ² (коэф. детермінації)	MAPE (%)
SVR	0.112	0.157	0.930	6.8
RFR	0.098	0.141	0.944	5.9
XGBoost Regressor	0.091	0.136	0.951	5.2

Як видно з результатів (табл.1), найкращі показники продемонструвала модель XGBoost Regressor. Вона забезпечила найнижчі значення MAE та RMSE, а також найвищий коефіцієнт детермінації ($R^2 = 0.951$). Додатково важливим показником є MAPE, який склав лише 5.2%, що свідчить про високу відносну точність прогнозу навіть за умови коливань у структурі споживання. За цільовий параметр прогнозування обрано споживання електроенергії (power_kwh). Порівняно з іншими моделями:

- Support Vector Regressor (SVR) показав позитивні результати на короткострокових прогнозах, проте мав вищу відносну похибку (MAPE 6.8%) у порівнянні з ансамблевими методами.
- Random Forest Regressor (RFR) продемонстрував стабільність та кращу

точність від SVR за всіма метриками, зокрема, зменшив середню похибку до MAE = 0.098 кВт та MAPE = 5.9%.

- XGBoost Regressor перевершив обидві моделі, забезпечивши найбільш збалансоване поєднання високої точності та узагальнювальної здатності.

Тож результати експериментів підтвердили доцільність використання більш складних ансамблевих методів для задач прогнозування енергоспоживання. Завдяки здатності враховувати нелінійні залежності та ефективно працювати з великими обсягами даних, XGBoost може бути рекомендований як базова модель для практичних систем прогнозування у сфері «розумних мереж» та управління енергоспоживанням.

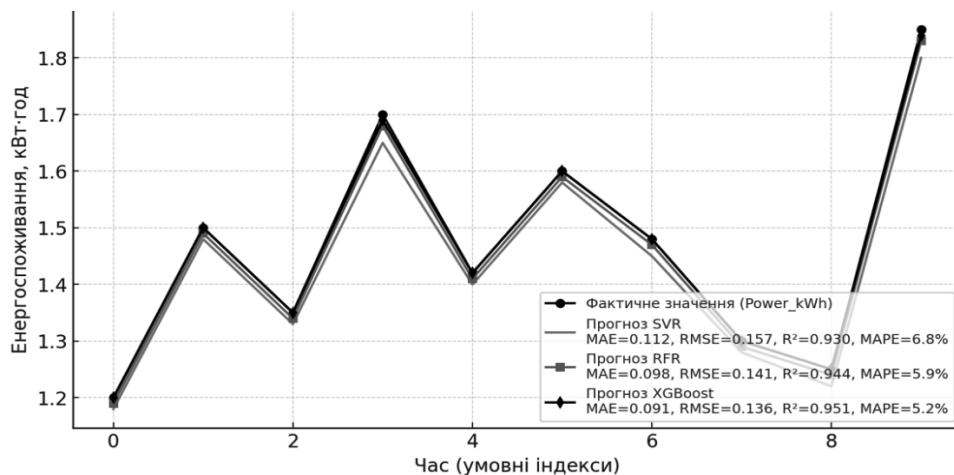


Рис. 3. Порівняння прогнозів моделей регресії (SVR, RFR, XGBoost)

5. Класифікація режимів роботи

У другому етапі дослідження вирішувалася задача класифікації режимів роботи енергоспоживання (device_mode), яка передбачала приналежності кожного відрізка часу до однієї з трьох категорій:

- низький режим (low) — відповідає періодам мінімального базового навантаження, зазвичай у нічний час або у випадку тривалої відсутності мешканців;
- середній режим (medium) — характеризує типовий денний рівень споживання у робочі дні, коли викорис-

товуються побутові прилади середньої потужності;

- піковий режим (peak) — відображає короточасні стрибки навантаження, що виникають у вечірні години або у моменти одночасної роботи кількох енергоємних пристроїв (наприклад, кондиціонера, пральної машини та кухонної техніки).

Для розв'язання задачі було застосовано алгоритмічні підходи:

- Random Forest Classifier — ансамблевий метод на основі дерев рішень, стійкий до шуму та добре працює з некорельованими ознаками;

- SVM Classifier — метод опорних векторів із використанням RBF-ядра, здатний будувати нелінійні межі рішень для складних даних подальшого порівняння;
- XGBoost Classifier — алгоритм градієнтного підсилення над деревами рішень, що продемонстрував високу гнучкість у роботі з великими та частково зашумленими даними. Його застосування виявилось доцільним з огляду на складність розподілу навантаження та можливу наявність нелінійних залежностей у даних.

Навчання та тестування обох моделей здійснювалося на тих самих вибірках, що і в задачі регресії: 70% даних було використано для навчання, 30% — для тестування (параметр `random_state=42` забезпечив відтворюваність результатів).

Якість класифікації оцінювалася за стандартними метриками: точність (Accuracy), точність передбачення (Precision), повнота (Recall) та F1-міра. Отримані результати подано у таблиці 2.

Таблиця 2.

Метрики класифікації режиму електроспоживання розумного дому

Модель	Accuracy	Precision	Recall	F1-score
Random Forest	0.894	0.901	0.889	0.894
SVM Classifier	0.881	0.887	0.873	0.879
XGBoost Classifier	0.912	0.918	0.905	0.911

Аналіз показує, що XGBoost Classifier знову перевершує інші моделі за всіма показниками. Водночас Random Forest показав кращі результати за метриками порівняно з SVM, що підтверджує його здатність ефективно працювати із зашумленими даними (Рис.4).

Зокрема, точність і повнота у XGBoost Classifier перевищили 91%, що підтверджує його придатність для впровадження у системах розумного дому. Це відкриває можливості для задач балансування енергоспоживання, попередження перевантажень електромережі та побудови адаптивних тарифних стратегій.

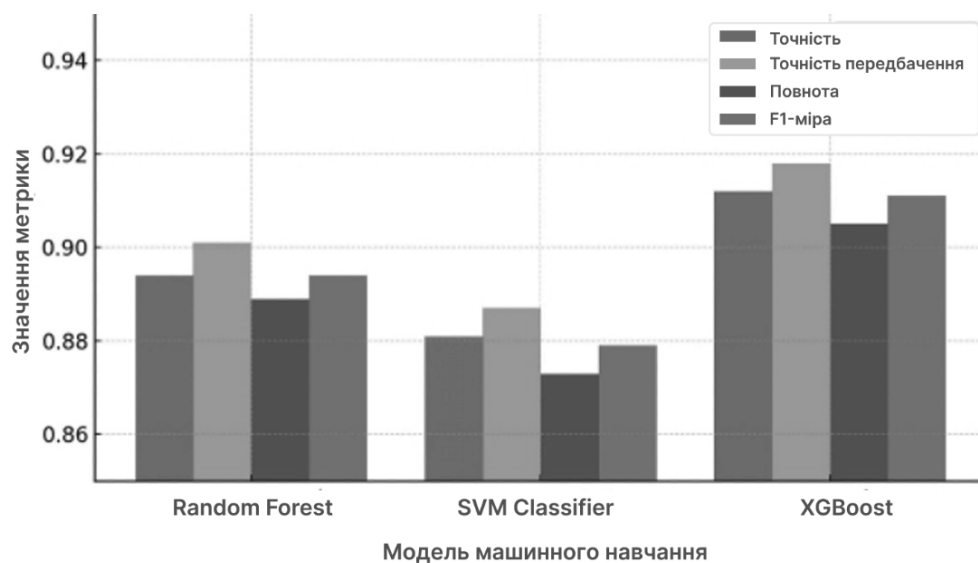


Рис. 4. Порівняння моделей класифікації режимів роботи

6. Кластеризація профілів енергоспоживання

На третьому етапі дослідження було проведено кластеризацію добових профілів

енергоспоживання з метою виявлення груп днів із подібною структурою навантаження. Для цього вихідні дані PSML були агреговані до добового масштабу та нормалізовані, що дало можливість зосередитися

на формі профілю, а не на абсолютних величинах споживання.

Першим було застосовано алгоритм KMeans, який належить до класу методів розбиття даних на заздалегідь визначену кількість кластерів. Для вибору оптимального значення параметра k використовували метод «лікоть» та коефіцієнт силуету, що дозволило визначити три стійкі групи профілів. Перша група відповідала типовим будням із вираженими піками споживання у ранкові та вечірні години, друга – вихідним дням із більш рівномірним та згладженим навантаженням, а третя група відображала аномальні випадки, коли споживання електроенергії мало нетипово високі або низькі значення протягом доби. Таким чином KMeans дозволив сформувати структуровану типологію днів за їхніми характерними режимами роботи побутових пристроїв.

Паралельно було використано алгоритм DBSCAN, який, на відміну від KMeans, не потребує попереднього визначення кількості кластерів. Цей метод показав себе ефективним у пошуку щільних зон подібних профілів та автоматичного виявлення «шумових» точок, що не належали жодному кластеру. Як результат, DBSCAN виявив компактні групи днів із подібною поведінкою споживання, а також виділив окремі аномалії, які відображають нетипові сценарії використання електроенергії в системах розумного дому.

Порівняльний аналіз продемонстрував, що KMeans є більш придатним для опису повторюваних і структурованих шаблонів енергоспоживання, тоді як DBSCAN надає кращі результати у випадках пошуку рідкісних або відхилених від норми режимів. Відповідні результати візуалізовано на рисунку 5, де відображено типові кластери, отримані за допомогою обох методів. Для кожного кластера побудовано середній профіль споживання, який обчислюється як середнє значення добового споживання для всіх днів, що належать до кластера. Альтернативно, може використовуватися медіанний профіль, щоб зменшити вплив аномалій. Саме ці середні або медіанні графіки показані на рисунку 5, і вони ілюструють типову форму навантаження для кожного кластера.

Отримані кластери мають важливе прикладне значення. Вони можуть бути використані для персоналізації прогнозів залежно від типу дня, оптимізації графіків роботи енергоємних приладів, а також для розробки більш гнучких та справедливих тарифних стратегій, що враховують різні моделі споживання електроенергії. Таким чином, кластеризація дозволяє не лише виявити приховані закономірності у даних, а й забезпечує основу для практичного застосування в системах розумного дому та енергетичному менеджменті.

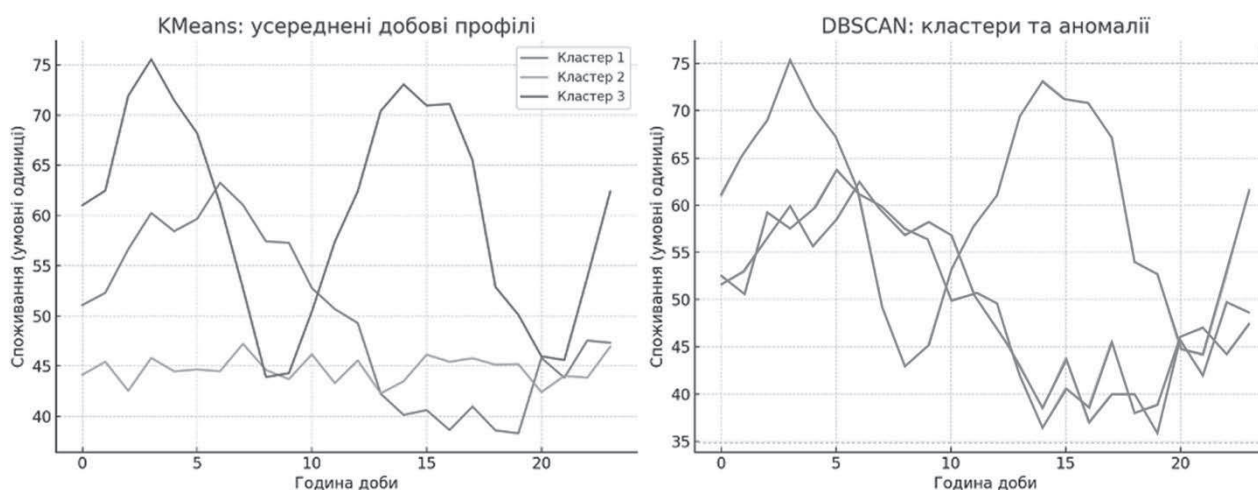


Рис. 5. Результати кластеризації профілів енергоспоживання (KMeans та DBSCAN)

Висновки

У статті досліджено можливості застосування методів машинного навчання для аналізу та прогнозування енергоспоживання в системах розумного дому на основі датасету PSML. Було реалізовано комплексний підхід, що включає три основні етапи: регресійне прогнозування, класифікацію режимів роботи та кластеризацію профілів споживання.

На етапі регресії порівнювались моделі SVR, RFR, XGBoost Regressor. Найкращі результати показав XGBoost Regressor, що забезпечив найнижчі похибки ($MAE = 0.091$, $RMSE = 0.136$, $MAPE = 5.2\%$) та найвище значення коефіцієнта детермінації ($R^2 = 0.951$). Це свідчить про високу здатність моделі враховувати нелінійні залежності та узагальнювати дані, що робить її найбільш придатною для задач прогнозування енергоспоживання.

Етап класифікації дозволив виділити три основні режими роботи: низький, середній та піковий. Порівняння моделей показало, що XGBoost Classifier також перевершив Random Forest та SVM за всіма метриками, досягнувши точності понад 91%. Це підтверджує його ефективність для задач віднесення споживання до різних режимів (низький, середній, піковий), що є критично важливим для балансування навантаження та запобігання перевантаженням мережі.

Кластеризація добових профілів споживання із застосуванням KMeans та DBSCAN дозволила виділити типові будні, вихідні та аномальні дні. KMeans виявив повторювані шаблони, тоді як DBSCAN ефективно ідентифікував рідкісні або нетипові сценарії, що забезпечує можливість адаптивного планування роботи енергоємних пристроїв та розробки стратегій тарифікації.

Завдяки комплексному застосуванню методів регресії, класифікації та кластеризації досягнуто:

- точного прогнозування енергоспоживання;
- коректної ідентифікації режимів роботи пристроїв;
- виділення типових та аномальних сценаріїв споживання.

Узагальнюючи результати, можна зробити висновок, що поєднання різних підходів машинного навчання дозволяє не лише досягти високої точності прогнозування енергоспоживання, а й гнучко управляти режимами роботи пристроїв та виявляти приховані закономірності у даних. Найбільш ефективним алгоритмом у задачах як регресії, так і класифікації виявився XGBoost, що робить його перспективною базовою моделлю для впровадження у практичні системи розумного дому та розумних мереж.

References

1. Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice* (2nd ed.). OTexts. <https://otexts.com/fpp2/>
2. Weron, R. (2014). Electricity price forecasting: A review of the state-of-the-art with a look into the future. *International Journal of Forecasting*, 30(4), 1030–1081.
3. Ryu, S., Noh, J., & Kim, H. (2017). Deep neural network based demand side short term load forecasting. *Energies*, 10(1).
4. Open source PSML dataset <https://github.com/tamu-engineering-research/Open-source-power-dataset?tab=re-adme-ov-file>
5. Sinitsyn, I. P., Shevchenko, V. L., Doroshenko, A. Yu., & Yatsenko, O. A. (2025). Research of software solutions for forecasting electricity production and consumption in Ukraine based on machine learning methods. *Problems of programming and information technologies*, 15(1), 45–58. <https://pp.isoftware.kiev.ua/ojs1/article/view/586>
6. Python Software Foundation. (2024). *Scikit-learn: Machine learning in Python*. <https://scikit-learn.org>
7. Vyshnevskyy, O. V., & Zhuravchak, L. (2023). Machine Learning Methods to Increase the Energy Efficiency of Buildings. *Computer Systems and Networks*, 14, 189–209. https://www.researchgate.net/publication/378031111_Machine_Learning_Methods_to_Increase_the_Energy_Efficiency_of_Buildings
8. Li, W., & Wang, J. (2019). Electricity consumption forecasting using a hybrid model based on data preprocessing and long short-

- term memory neural network. Energy, 169, 1099–1110.
9. Ahmed, R., & Khalid, M. (2020). A review on the selected applications of forecasting models in renewable energy systems. Renewable and Sustainable Energy Reviews, 124, 109792.
10. Khac, T. N., & Bui, T. V. (2021). Load forecasting using machine learning: A comparative study of neural networks and random forests. Energy Reports, 7, 394–403.

Додаток 1.

Фрагмент програмного коду для прогнозування енергоспоживання в системі розумний дім

```
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
```

```
# 1. Завантаження та попередня обробка даних
df = pd.read_csv("psml_data.csv",
parse_dates=["datetime"])
df = df.dropna(subset=["load"]) # видалення рядків з відсутнім споживанням
```

```
# 2. Вибір ознак та цільової змінної
```

```
features = ["temperature", "humidity", "wind_speed"] # метео-параметри як ознаки
target = "load" # споживання енергії
```

```
X = df[features]
y = df[target]
```

```
# 3. Розділення даних на тренувальні та тестові множини
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

```
# 4. Ініціалізація та навчання моделі Random Forest
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

```
# 5. Прогнозування та оцінка точності моделі
y_pred = model.predict(X_test)
rmse = mean_squared_error(y_test, y_pred, squared=False)
print(f"RMSE на тестових даних: {rmse:.2f} кВт·год")
```

Одержано: 26.09.2025

Внутрішня рецензія отримана: 13.10.2025

Зовнішня рецензія отримана: 04.10.2025

Про авторів:

Гайдукевич Владислав Олегович,
аспірант

<https://orcid.org/0000-0002-0614-6778>

Дорошенко Анатолій Юхимович,
доктор фізико-математичних наук,
професор,
завідувач відділу теорії комп'ютерних
обчислень ІПС НАН України,
професор кафедри інформаційних систем
та технологій КІІ імені Ігоря Сікорського.
<http://orcid.org/0000-0002-8435-1451>,

Місце роботи авторів:

Інститут програмних систем НАН України,
03187, м. Київ-187, проспект Академіка
Глушкова, 40.
e-mail: doroshenkoanatoliy2@gmail.com,
gaidukevichvlad@gmail.com