scientific reports

Check for updates

OPEN

High-throughput behavioral screening in *Caenorhabditis elegans* using machine learning for drug repurposing

Antonio García-Garví & Antonio-José Sánchez-Salmerón✉

Caenorhabditis elegans **is a widely used animal model for researching new disease treatments. In recent years, automated methods have been developed to extract mobility phenotypes and analyse, using statistical methods, whether there are differences between control strains and disease model strains. However, these methods present certain limitations in detecting subtle and non-linear patterns. In this study, we propose a high-throughput screening method based on machine learning, using classifiers that provide a recovery percentage as a measure of treatment effect. We evaluate two main approaches: traditional machine learning models based on behavioral features extracted from the worm's skeleton using Tierpsy Tracker, and deep neural networks that directly analyse video sequences. The results indicate that a Random Forest classifier trained with features extracted by Tierpsy Tracker offers higher accuracy and explainability, making it more suitable than deep learning models for drug testing experiments. Finally, to assess the applicability of our method, we processed data from a published drug repurposing study on** ***unc-80*** **mutants based on statistical methods. The results highlight the potential of machine learning models to enhance automated phenotypic screening in animal models, providing a more robust and quantitative evaluation of treatment effects by considering more complex and subtle patterns.**

Keywords C. elegans, Machine learning, Computational ethology, Phenotypic screen, Drug screening, Disease models

Currently, many diseases lack an approved treatment. This issue is even more pronounced in rare diseases, which, due to their low prevalence, receive less attention and fewer research resources[1]. Additionally, new diseases are discovered every year, further expanding this challenge[2]. To accelerate drug discovery, researchers turn to model organisms such as *Caenorhabditis elegans* (*C. elegans*). This nematode has a fully sequenced, simple genome[3], with 60-80% of its genes homologous to those in humans[4]. It can be genetically manipulated relatively easily using techniques such as RNA interference (RNAi) and CRISPR/Cas9 gene editing. Its small size (1 mm in length) and short lifespan (2–3 weeks) allow for large-scale experiments at a low cost. *C. elegans* has proven to be a versatile and cost-effective tool for modeling diseases such as Alzheimer's[5], Parkinson's[6] and amyotrophic lateral sclerosis (ALS)[7].

Manual analysis of *C. elegans* behaviour for large-scale phenotypic screening is a laborious and time-consuming process. For this reason, in recent years, video capture devices and algorithms have been developed that are capable of tracking and extracting features automatically[8–14].

Recently, a drug repurposing method was proposed for *C. elegans* models of Mendelian human diseases[15]. The authors use the CRISPR gene-editing technique to create a model of the disease they aim to study. Subsequently, screening assays are performed by treating the diseased worms with an FDA-approved drug library using a high-throughput imaging platform[14]. This platform captures high-resolution images and extracts a series of morphological, postural, and movement-related features[13]. Using block permutation t-tests with Benjamini-Yekutieli correction for multiple comparisons, they determine how many features show statistically significant differences compared to the control strain.

In an initial screening of 743 drugs with few replicates, they identify 30 potential hits, prioritizing those that shift three core features toward wild-type levels. In a second confirmation assay, more replicates are performed

Instituto de Automática e Informática Industrial, Universitat Politècnica de València, Camino de Vera S/N, 46022 Valencia, Spain. ✉email: asanchez@isa.upv.es

with the hits selected in the previous screening, and the compounds that still rescued the core disease model phenotype and did not exhibit many side effects are identified.

Although this approach has potential, it presents limitations. The manual selection of a small set of three core features to evaluate drug efficacy may overlook relevant information. In fact, when the analysis was expanded to 256 predefined features, no hits were detected, suggesting that the use of statistical methods such as block permutation t-tests with Benjamini-Yekutieli correction might be limiting the detection of subtle patterns. These methods, while interpretable, have low statistical power when correcting for multiple comparisons and generate binary results (significant p-values or not), without reflecting the magnitude of the differences. Moreover, they are designed to detect linear relationships, which may be insufficient for capturing complex interactions between multiple features.

In recent years, machine learning (ML) methods have demonstrated their ability to find complex patterns. In this work, we propose using a machine learning-based classifier whose output (confidence values) represents a recovery percentage, thus providing a quantitative measure of the treatment effect. Unlike traditional statistical methods, machine learning models have the capacity to detect subtle and non-linear patterns, as well as complex interactions between multiple features, offering a more powerful alternative for phenotypic data analysis. However, these techniques also present challenges, such as higher computational cost, interpretability issues, and the risk of overfitting.

In this work, we explore two approaches: (1) using features extracted from the worm's skeleton as input to a classifier based on traditional machine learning methods (random forest, XGBoost, logistic regression); (2) using deep learning (DL) models that take the captured videos as input[16]. This option has the advantage of not requiring skeletonisation of the worm initially, which can be prone to errors in cases of complex postures and overlaps. On the other hand, it allows for the exploration of other features beyond those extracted by current trackers. The results obtained showed a slight superiority of traditional machine learning classifiers, with Random Forest being the best among them. To evaluate the applicability of our method, we processed the data from the repurposing experiment by O'Brien et al.[15] and compared the results obtained with those from the statistical approach used in that work. Our findings suggest that the use of machine learning enables a more robust and quantitative assessment of the treatment effects by considering more complex and subtle patterns.

Methods

Datasets

The dataset generated in the study of O'Brien et al.[15] was used for this study. This dataset consists of 3 experiments, which we refer to throughout this manuscript as strain classification, initial drug screening, and confirmation drug screening. In the strain classification experiment, 25 *C. elegans* strains with mutations modelling rare diseases in humans were phenotyped and compared to the control strain N2. In the initial drug screening experiment, the authors performed a drug repurposing screen of 743 FDA-approved compounds to identify drugs that improve the behavioural phenotype of *unc-80* loss-of-function mutants. In the confirmation drug screening experiment, the repositioning screen is repeated using the most promising candidate compounds from the initial screen. Details of the experiments (mutant generation and assay protocol) can be found in their paper[15].

The videos were acquired using an automatic capture system[14]. They have a resolution of 12.4 µm/px and a frame rate of 25 fps. Each video contains 16 square wells, with approximately 3 worms in each well, as shown in Fig. 1. The videos were recorded sequentially for 3 periods: a 5-minute prestimulus, a 6-minute video with blue light stimulation (10 s pulses at 60, 160 and 260 s) and a 5-minute post-stimulus period. For our study, we only used the second period with blue light stimulation as it allows to detect more easily the differences between the different strains[14].

Quantitative drug screening method using a recovery index

This work proposes an alternative to the traditional approach of high-throughput screening for drug testing in animal models. The schematic representation of the method is shown in Fig. 2.

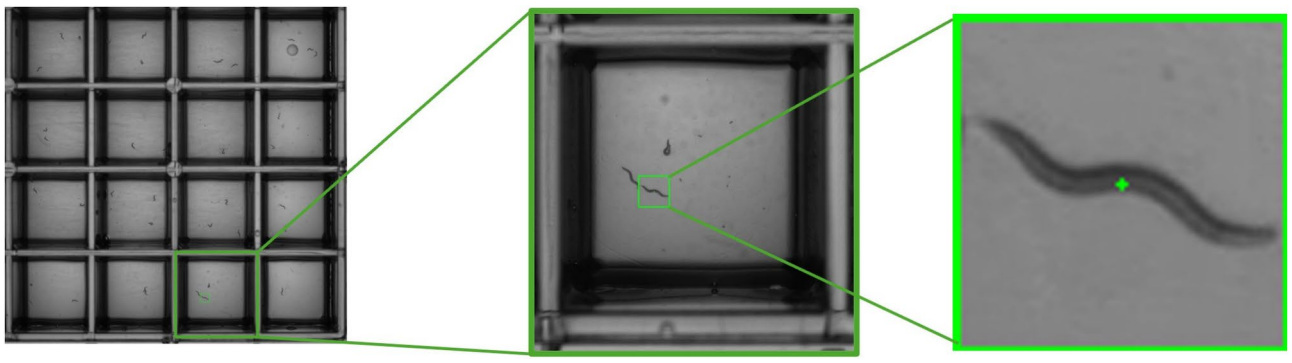

Fig. 1. Example of a video from the dataset used. From left to right: a frame from an original video in the dataset, a cropped image of a well, and a cropped image of a worm.

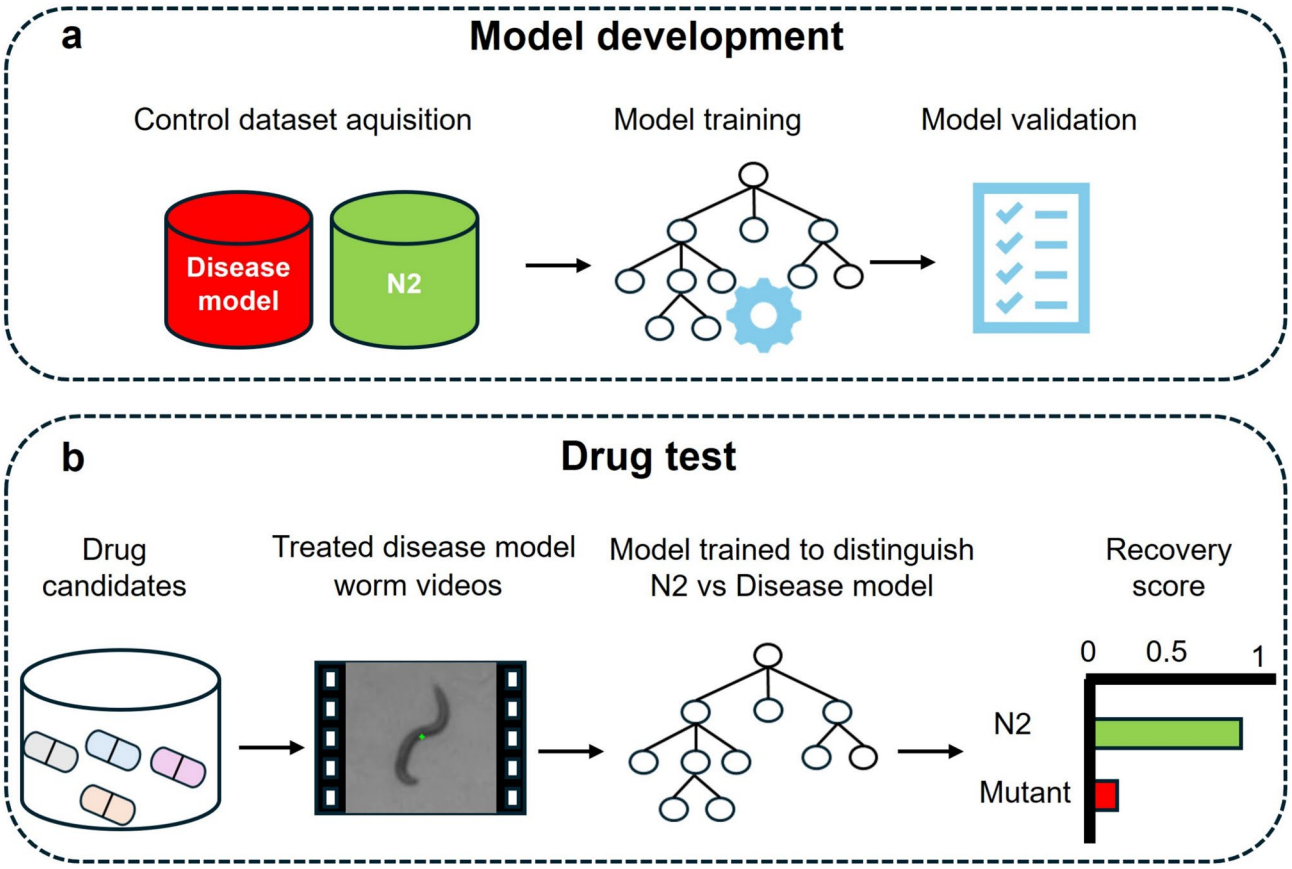

Fig. 2. Method outline.

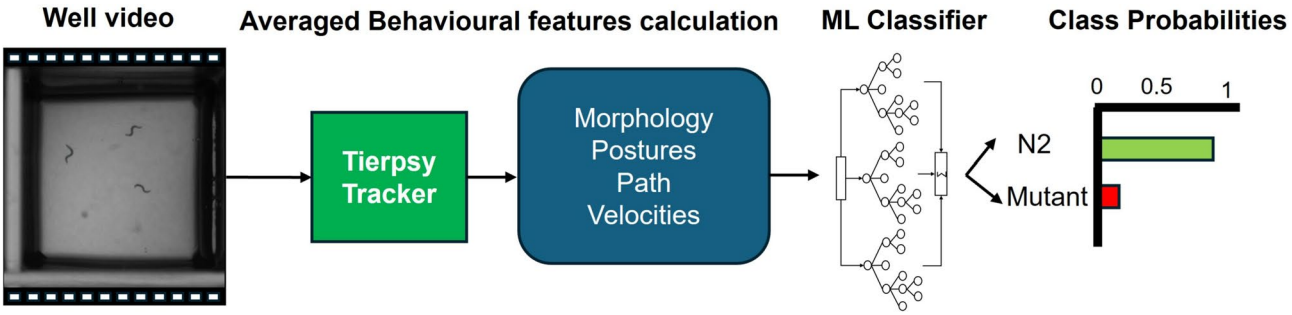

Fig. 3. Machine Learning Classification Pipeline. First, the average behavioural features per well are calculated using Tierpsy Tracker. These features are then used to perform classification with a machine learning model.

First (Fig. 2a), a model is trained to distinguish between the control strain N2 (healthy) and the disease model strain. It is crucial to ensure that the model can accurately differentiate between both strains, which is validated using an independent dataset. Once trained and validated, the model is then used to classify videos of *C. elegans* disease models treated with different drugs (Fig. 2b). The confidence values from the classifier's output are used as a recovery index: the greater the similarity between treated worms and the control strain, the higher the estimated recovery percentage. This index provides an automated and quantitative evaluation of the treatments' effects on the worms.

There are different alternatives for developing the classification model. In this paper we have analysed two: (1) classification from behavioural features extracted from the worm skeleton and (2) using deep neural networks to directly analyse video sequences of individual worms.

Behavioural feature-based classification with traditional machine learning

This method (Fig. 3) involves classification based on a set of features extracted from the worms' skeletons. To obtain these features, we used the Tierpsy Tracker software[13,17] which provides measurements related to speed

(e.g., average speed, maximum speed), morphology (e.g., length, curvature, area), and locomotion patterns. Tierpsy Tracker initially calculates features for each trajectory. However, it does not resolve cases of worm overlapping, and trajectories may sometimes be lost when worms approach the edge of the plate. In such cases, a new identity is assigned. As a result, instead of using individual trajectories, the final output is an average feature vector per well. For classification, we used traditional machine learning classifiers, including Logistic Regression, Random Forest, and XGBoost.

Individual worm classification using deep learning

In this method, we focus on classifying individual *C. elegans* trajectories from video sequences using artificial neural networks, building on our previous work[16]. This requires processing the videos of each well (Fig. 4a), detecting the worms, and tracking their trajectories. In our case, we used the trajectories obtained by Tierpsy Tracker to determine the centroid of the worm in each frame. From the centroid, we generate the videos by creating an 80x80 pixel window centered on the worm, ensuring that the worm is fully visible within each window.

Due to memory limitations when processing long-duration videos, we used an approach that divides the videos into 30-second sub-videos (Fig. 4b). The final classification of each full video is obtained using a soft voting method, in which the confidence values returned by the artificial neural network for each sub-video are averaged. The class with the highest average probability value is selected as the final prediction.

We use as a starting point the CNN-Transformer model proposed in our previous work[16]. In that study, we hypothesised that the model could be improved by incorporating information related to velocity and displacement, since the videos are centred on the worm. Therefore, in this work, we introduced a position variation matrix that captures the changes in the worm's centroid position between consecutive frames. Based on this information, we modified the previous architecture and proposed a new bimodal model that, in addition to analyzing the videos, integrates the information contained in the displacement matrix, thereby improving the representation of *C. elegans* movement (Fig. 5).

The bimodal architecture consists of two main branches. A convolutional neural network (CNN) based on ResNet50[18] is responsible for extracting the visual features from each frame. The CNN was initialised with pre-trained ImageNet weights[19], enabling it to leverage prior knowledge gained from large visual datasets. Simultaneously, a fully connected network transforms the sequence of displacement vectors (2D coordinates) from the worm's trajectories into feature vectors of the same size as the visual features. For each timepoint, the visual and displacement features are concatenated, preserving the temporal structure of the sequence. These per-frame combined features are used as input tokens to a Transformer Encoder[20], where each token represents a distinct timepoint. Learnable positional embeddings are added to encode temporal information, and the resulting sequence is passed to a Transformer Encoder, which models temporal dependencies across frames. Finally, an additional fully connected network performs the classification into two classes: wild-type and mutant.

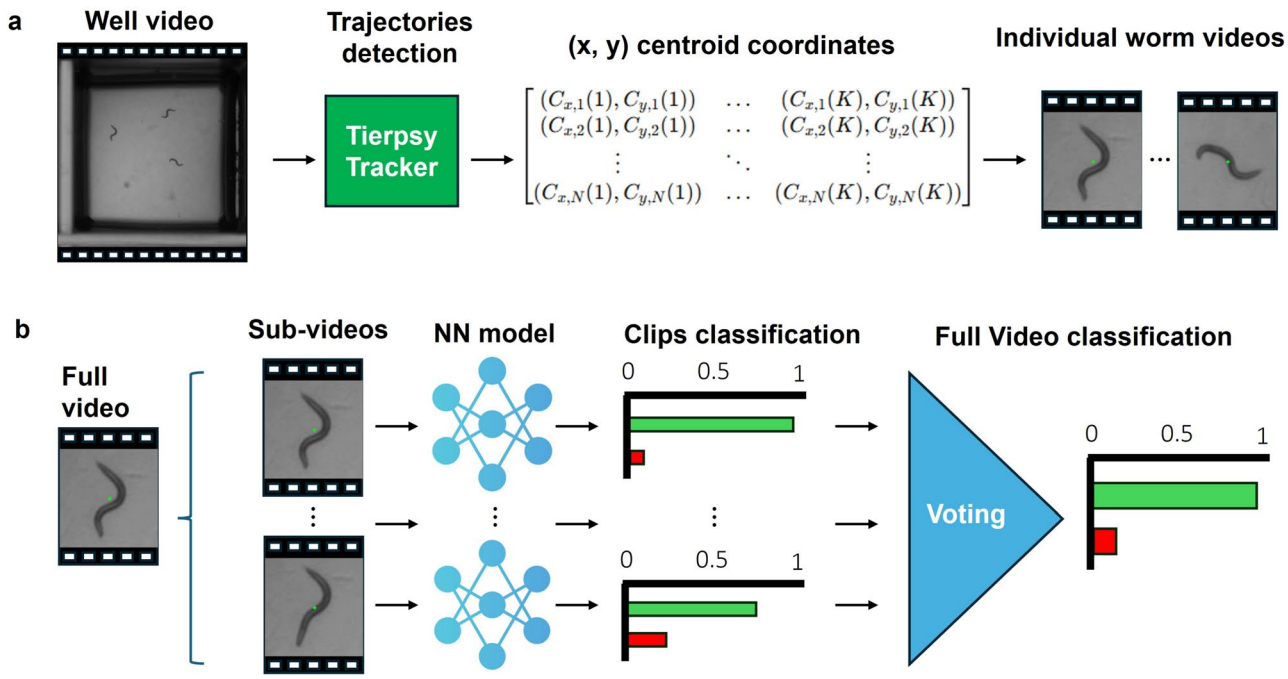

Fig. 4. Deep Learning-Based Classification Pipeline. First, videos of individual worms are extracted. Next, the full-length videos are divided into short clips. These short clips are then classified by the neural network. Finally, the overall classification is determined using soft voting.

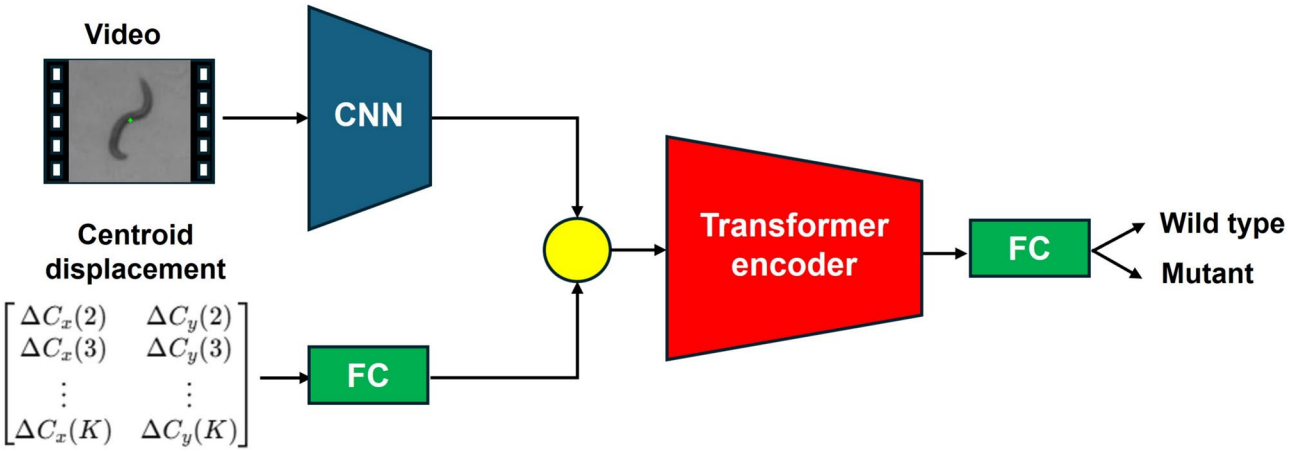

Fig. 5. Schematic representation of the proposed bimodal architecture. The model consists of two main branches: a CNN for extracting visual features, and a fully connected network for processing displacement vectors. Both branches are combined and passed through a Transformer Encoder to capture temporal relationships, followed by a fully connected layer for classification.

Comparison method of ML vs DL approaches

To compare the results obtained between the feature-based method and the individual worm video-based method, we decided to evaluate classification accuracy per well, as the Tierpsy Tracker results are obtained in this way, as discussed in the ML method section. Therefore, to calculate the artificial neural network prediction per well, the confidence values of all worms from the same well were averaged. We used the first dataset mentioned in the dataset section (strain classification) to compare accuracy in distinguishing between the N2 and *unc-80* strains. To compare the methods, we created a dataset of individual videos with a train/validation/test split such that:

- The number of frames in the trajectory is greater than 4500 (3 minutes).
- The number of sub-videos per trajectory is the same (6 videos of 30 seconds).
- The number of trajectories in each class (control and disease model) is the same.
- Trajectories from the same well were placed in the same train/val/test set to prevent data leakage.

After this split, the training dataset consisted of 352 trajectories per class, and the test dataset had 152. When splitting the trajectories into 30 seconds clips, the deep learning model dataset consisted of 4224 clips for training and 1824 for test. Similarly, the dataset for evaluating the machine learning models was generated using the features of the wells corresponding to the videos utilised in the deep learning model, resulting in 160 wells per class for training and 69 wells per class for test. Using the scikit-learn library[21], the precision (Eq. 1), recall (Eq. 2) and F1-score (Eq. 3) metrics are computed for each class. Finally, the scores for both classes are averaged to obtain the overall performance metrics.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{1}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2}$$

$$\text{F1 score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

Results

Feature-based classification: model selection and evaluation

In this experiment, Logistic Regression, Random Forest, and XGBoost were evaluated for classification based on well averaged features extracted from worm trajectories. Before training the models, the features underwent preprocessing: (1) Features with more than 5% missing values (NaN) were filtered out, and (2) the remaining NaN values were imputed using the mean.

Next, the most relevant features were selected using the Recursive Feature Elimination (RFE) method with initial cross-validation (nfolds = 5) and the default parameters of the models in their scikit-learn library implementation[21]. With the selected features, hyperparameter optimization was performed using a random grid search within a predefined range for each model. The values used are shown in Supplementary Table S1, S2, and S3. Performance metrics obtained from 5-fold cross-validation are reported in Supplementary Table S4. Finally, the best model was evaluated with the test dataset. The results of precision, recall, and F1-score metrics are presented in Table 1.

The most accurate model was Random Forest, although the results were similar across all models analysed.

Model	Precision	Recall	F1-Score
Logistic Regression	0.95	0.95	0.95
Random Forest	0.96	0.96	0.96
XGBoost	0.94	0.94	0.94

Table 1. Performance metrics of traditional ML models for well-based classifications on strain classification dataset.All metrics are computed on the held-out test set.

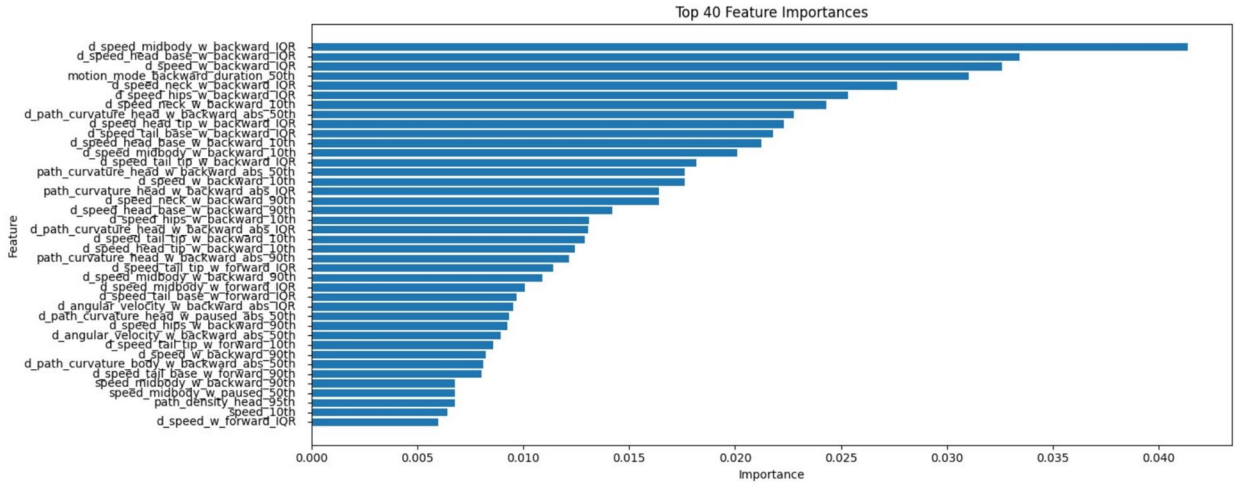

Fig. 6. Top 40 Feature importance ranking of the random forest model.

Neural network training and justification of bimodal model

The neural network models were implemented and trained using the Pytorch[22] deep learning framework for 20 epochs using the cross-entropy loss cost function and Adam's optimizer. A learning rate scheduler (initial learning rate of 1e-5) was used which reduces the learning rate by a factor of 10 when 3 epochs pass without reducing the value of the loss function in the validation set. The batch size chosen was 6 samples considering our memory constraints. Data augmentation included rotations and adjustments to brightness, saturation, and contrast. Early stopping was applied to prevent overfitting. The input size to the network was 224x224 pixels, so the images were rescaled to that size. Given memory limitations, 30-frame clips at 1 fps were used, which had shown to be an appropriate resolution in our previous work[16]. The hardware used to train the model includes a Ryzen 9 3900X processor with 12 cores running at 3.8 GHz, 128GB of DDR4 3200MHz memory, and a Nvidia RTX3090 GPU with 24GB of DDR4 memory.

To estimate performance variability, we additionally trained the model using multiple train/validation splits within the training data . The results (Supplementary Table S5) showed slightly lower performance than the held-out test set, which we attribute to the smaller training set size per split and the use of shorter training schedules due to computational constraints.

To confirm our hypothesis on the need to incorporate velocity-related features and, therefore, employ a bimodal model, we conducted three experiments. First, we compared the accuracy of the machine learning model when only non-velocity-related features were used. The F1-score dropped from 96% to 78%. Secondly, we trained a deep learning model by removing the branch containing displacement information. The bimodal neural network obtained a 93% f1-score compared to 86% for the artificial neural network trained with only the videos as input. Thirdly, we analysed the key features influencing the random forest model prediction based on Gini impurity (Fig. 6), finding that most are related to velocity.

ML vs DL performance comparison on classification per well

The results indicate that the machine learning approach outperforms the deep learning model on the held-out test dataset. Specifically, the ML model achieves higher macro-averaged precision (96% vs. 93%), recall (96% vs. 93%), and F1-score (96% vs. 93%) compared to the DL model. The corresponding confusion matrices are shown in Fig. 7.

Although the deep learning model achieves lower performance, it remains promising given the typical data limitations associated with such models. By examining the videos where the network frequently misclassifies, we identified key challenges, including worm overlapping (Fig. 8a), worms near the edges (Fig. 8b), and some detection errors (Fig. 8c).

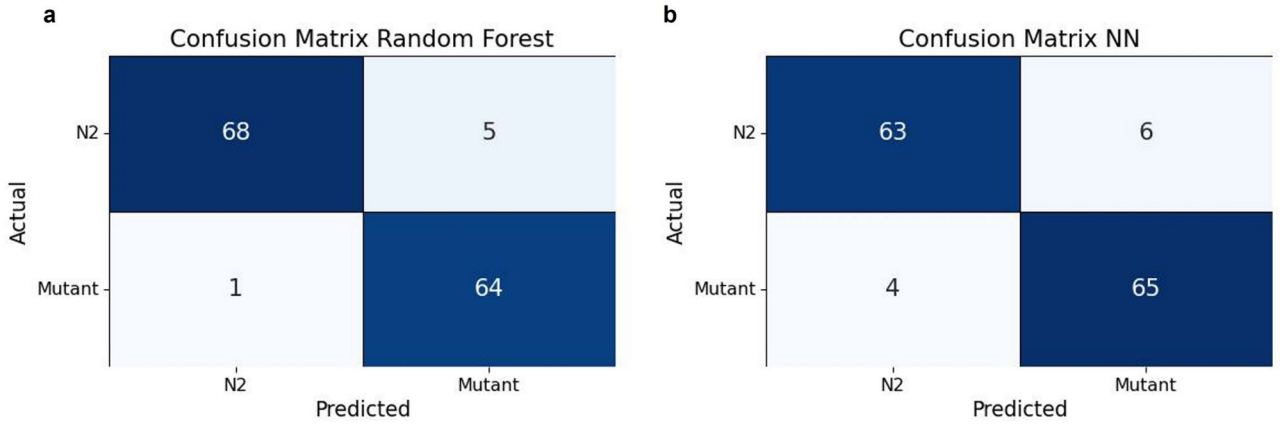

Fig. 7. Comparison of ML vs. DL method on strain classification dataset. Confusion matrices are based on the same held-out test set.

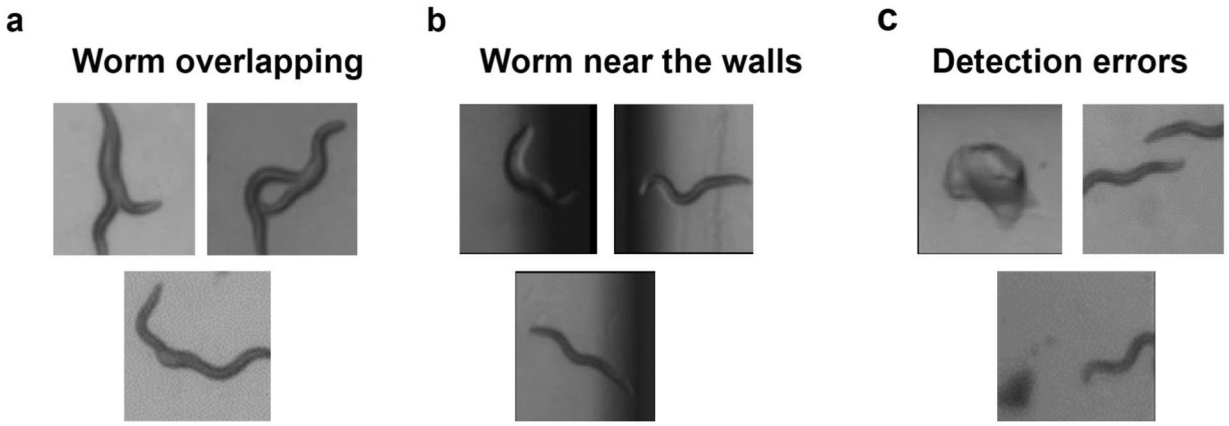

Fig. 8. Analysis of problem cases for the deep learning model, highlighting key challenges such as a) worm overlapping, b) worms near the edges, and c) detection errors.

	Precision	Recall	F1-score	Support
N2	0.75	0.83	0.79	60
glr-4	0.81	0.72	0.76	60

Table 2. Strain classification N2 vs *glr-4*. Classification results of the trained model on control samples in the test dataset.

Detecting subtle differences with machine learning in the *glr-4* mutant

To assess the ability of machine learning to detect subtle or complex phenotypic differences, we conducted an experiment on the *glr-4* mutant strain. In the original analysis by O'Brien et al.[15], *glr-4* was reported to exhibit no statistically significant differences from wild-type N2 worms across any of the 8,289 behavioral features.

We trained a Random Forest classifier to distinguish between N2 and *glr-4* samples using Tierpsy-derived features, following the same preprocessing and training pipeline described in the previous experiments. Despite the absence of significant individual features, the classifier achieved an F1-score of 0.77 on a held-out test set. The results of precision, recall, and F1-score metrics are presented in Table 2.

This result suggests that the behavioral phenotype of *glr-4* is distinguishable from wild-type N2 based on multivariate patterns, even when univariate statistical testing fails to detect any differences. It highlights a key advantage of machine learning approaches: the ability to identify subtle, distributed, or nonlinear patterns in the data that may not be captured by traditional statistical comparisons.

This case illustrates how classification-based phenotyping can complement standard statistical testing, offering enhanced sensitivity for detecting complex behavioral effects.

	Precision	Recall	F1-score	Support
N2	0.97	0.97	0.97	96
unc-80	0.97	0.97	0.97	96

Table 3. Initial drug screening. Classification results of the trained model on control samples in the test dataset.

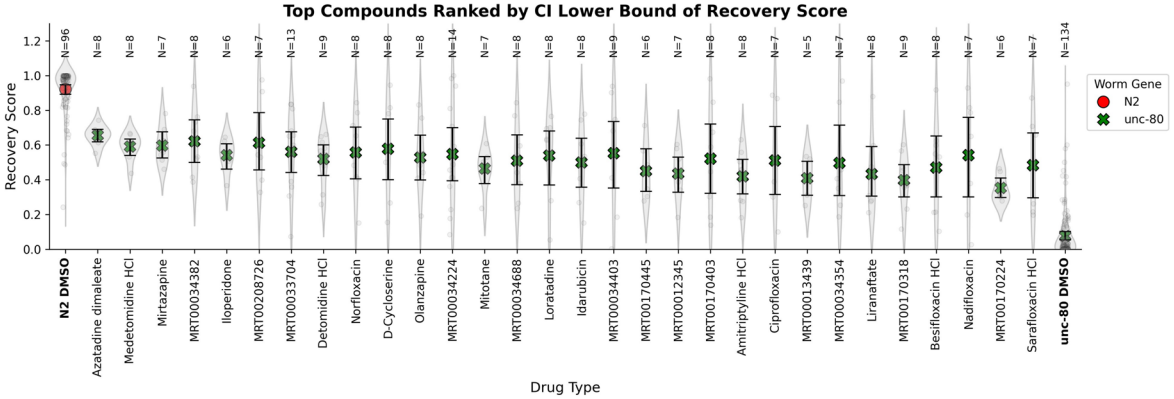

Fig. 9. Drug Repurposing Initial Screen. Recovery scores for top-ranking compounds based on the lower bound of the 95% confidence interval. The scatter plot displays mean recovery scores for each drug, categorised by worm genotype (N2 in red, *unc-80* in green), with error bars representing bootstrap-derived confidence intervals. A violin plot illustrates the distribution of individual recovery scores, and sample sizes (N) are annotated above each drug type. N2 DMSO and *unc-80* DMSO controls are highlighted in bold.

Drug repurposing screening

To evaluate the method proposed in this study, we processed the drug screening data from *unc-80* mutants conducted in O'Brien et al.[15]. Alongside the drug-treated *unc-80 worms*, the screening dataset also includes control samples: wild-type N2 worms and *unc-80* mutants, both treated with 1% DMSO. As mentioned earlier, this experiment consists of two phases: an initial screening, in which a library of 743 FDA drugs was tested with few replicates to select the most promising compounds, and a confirmation phase, in which more replicates were performed with the compounds selected in the previous phase. Among all the approaches analysed in this paper, the one that achieved the best results in distinguishing between the N2 and *unc-80* strains was using a Random Forest classifier based on the behavioural features extracted by Tierpsy Tracker. Therefore, we adopted this approach in this experiment. The procedure was as follows: (1) the model was trained exclusively on the control samples to distinguish between wild-type N2 worms and *unc-80* mutants (both treated with 1% DMSO) following the procedure described in in the ML method section; (2) the model's performance was validated on a held-out test set of control samples to ensure its ability to generalise to unseen data; (3) once trained and validated, the model was applied in inference mode to classify *unc-80* samples treated with different compounds.The hypothesis is that if a treatment is effective, the worm's behavior will shift toward the wild-type phenotype, and the model will assign a higher probability of belonging to the N2 class; (4) the compounds were ranked based on their recovery score, defined as the model-predicted probability of belonging to the N2 class. Since the number of replicates per compound is not constant due to factors such as well cancellation caused by precipitation, contamination, or tracking errors, the mean may not be a sufficiently robust measure. To mitigate this limitation and reduce the influence of potential random variations in the data, compounds were ranked using the lower bound of a 95% confidence interval, calculated via bootstrapping (n = 10,000 resamplings, with replacement). This strategy minimises the impact of potential biases arising from replicates with a lower number of samples.

Initial screening results

The model was trained using data from the control strains (N2 and *unc-80*, both treated with 1% DMSO). A total of 764 wells were used for training and 192 for test. The test results are presented in Table 3. As shown, the model achieved a high F1-score of 0.97. Next, the wells corresponding to the *unc-80* strain treated with each compound were classified, obtaining a confidence interval for each one. The compounds were ranked based on the lower bound of this interval to identify those with the highest recovery score. Figure 9 presents the 30 most promising compounds, along with the values obtained for the controls (N2 and *unc-80*). To enhance the robustness of the analysis, compounds with fewer than five samples were excluded.

Table 4 below compares the 30 compounds with the highest lower bound confidence interval (CI) calculated by the method proposed in this paper with the 30 hits chosen in the paper by O'Brien et al.[15] .

Common Hits		Recovery Score Hits		O'Brien et al.[15] Criteria Hits	
Compound	CI lower	Compound	CI lower	Compound	CI lower
Azatadine dimaleate	0.6180	MRT00034382	0.4996	Vinblastine	0.2926
Medetomidine HCl	0.5393	MRT00208726	0.4558	Clozapine	0.2845
Mirtazapine	0.5253	MRT00033704	0.4412	Ziprasidone hydrochloride	0.2808
Iloperidone	0.4610	MRT00034224	0.3936	Abitrexate	0.2368
Detomidine HCl	0.4234	MRT00034688	0.3701	Sulindac	0.2249
Norfloxacin	0.4048	MRT00034403	0.3516	Carbenicillin disodium	0.2070
D-Cycloserine	0.3995	MRT00170445	0.3325	Ofloxacin	0.1803
Olanzapine	0.3978	MRT00012345	0.3276	Moxifloxacin	0.1518
Mitotane	0.3766	MRT00170403	0.3209	Atorvastatin calcium	0.1169
Loratadine	0.3693	MRT00013439	0.3099	Fenofibrate	0.1129
Idarubicin	0.3568	MRT00034354	0.3088	Rizatriptan benzoate	0.1129
Amitriptyline HCl	0.3174	MRT00170318	0.3005	Mesalamine	0.0987
Ciprofloxacin	0.3143	Besifloxacin HCl	0.3002	Sulfadoxine	0.0961
Liranaftate	0.3048	Nadifloxacin	0.2998	Ivabradine HCl	0.0589
		MRT00170224	0.2974	Rofecoxib	0.0474
		Sarafloxacin HCl	0.2958	Daunorubicin HCl	0.0217

Table 4. Comparison of the 30 compounds with the highest recovery score calculated by the proposed method versus those chosen in the paper by O'Brien et al.[15]. The lower CI lower value is included for each compound.

	Precision	Recall	F1-score	Support
N2	1.0	0.94	0.97	17
unc-80	0.94	1.0	0.97	17

Table 5. Confirmation drug screen. Classification results of the trained model on control samples in the test dataset.

Our method identified additional promising compounds, including MRT00034382, MRT00208726, and MRT00033704, which were not highlighted in the previous study, suggesting that machine learning may capture overlooked phenotypic patterns. Drugs such as Daunorubicin HCl, Rofecoxib, and Ivabradine HCl, which were selected by O'Brien et al.[15], did not rank highly in our screening.

Confirmation screen results

Analogous to the initial screening, the model was trained and validated using data from the control strains (N2 and *unc-80*, both treated with 1% DMSO). A total of 134 wells were used for training and 34 for test. The test results are presented in Table 5.

As in the initial screening, inference was made on data from diseased worms treated with the compounds chosen in the first phase of the reference paper[15], as we do not have data from all hits identified by our method. A confidence interval was calculated for each compound and ordered according to the lower limit of this interval. The results are shown in Fig. 10. The criterion of excluding compounds with less than 5 samples was maintained to improve the robustness of the results.

The highest-ranked compounds by our model included Idarubicin, Azatadine Dimaleate, and Atorvastatin Calcium, with recovery scores of 0.49, 0.43, and 0.43, respectively. These compounds exhibited a moderate level of phenotypic recovery in *unc-80* mutants. The next analysis we conducted was to determine the number of reported side effects in[15]. Side effects are characteristics that show no significant difference between *unc-80* mutants and wild-type N2 worms treated with 1% DMSO, but exhibit a significant difference between *unc-80* mutants treated with the drug and N2 worms.

Let $F = \{f_1, f_2, \ldots, f_n\}$ denote the set of behavioral features extracted using Tierpsy Tracker, where $n = 8289$. For each feature $f_i \in F$, the following experimental groups are compared:

- G_{wt}: wild-type N2 worms treated with 1% DMSO
- G_{mut}: *unc-80* mutants treated with 1% DMSO
- $G_{\text{mut+drug}}$: *unc-80* mutants treated with a candidate drug

Each feature f_i is tested for statistical significance using a block permutation t-test with 100,000 permutations. Multiple comparisons are corrected using the Benjamini–Yekutieli method, with a significance threshold $\alpha = 0.05$.

A feature f_i is defined as exhibiting a side effect if:

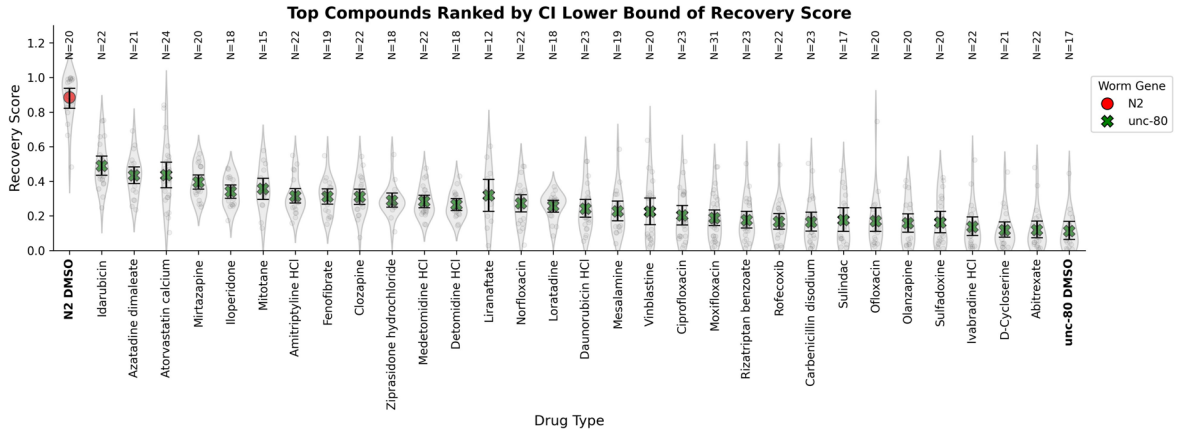

Fig. 10. Recovery scores for top-ranking compounds based on the lower bound of the 95% confidence interval. The scatter plot displays mean recovery scores for each drug, categorised by worm genotype (N2 in red, *unc-80* in green), with error bars representing bootstrap-derived confidence intervals. A violin plot illustrates the distribution of individual recovery scores, and sample sizes (N) are annotated above each drug type. N2 DMSO and *unc-80* DMSO controls are highlighted in bold.

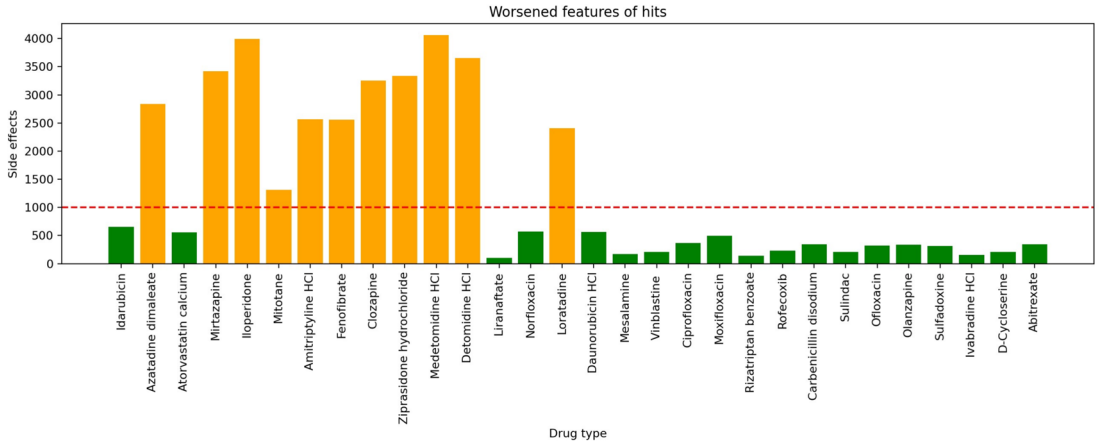

Fig. 11. Side effects reported in O'Brien et al. for the top-ranked compounds based on our selection criteria. Compounds are ordered according to their recovery score ranking. Bars are colored green for compounds with fewer than 1000 reported side effects and orange for those exceeding this threshold. The red dashed line marks the 1000 side effect cutoff.

$$\text{p-value}(G_{\text{mut}}, G_{\text{wt}}) \geq \alpha \quad \text{and} \quad \text{p-value}(G_{\text{mut+drug}}, G_{\text{wt}}) < \alpha$$

This condition identifies features where the drug induces a behavioral change that was not present in the untreated mutant, resulting in a significant deviation from the wild-type phenotype.

The number of side effects for a given treatment is defined as the number of features $f_i \in F$ satisfying the above condition.

These are shown in Fig. 11. Based on the criteria for side effects proposed in this study, Idarubicin and Atorvastatin would be interesting candidates, as they have a relatively low number of side effects (654 and 553, respectively). Azatadine, on the other hand, would not be a good candidate due to its 2840 worsened characteristics.

Discussion

The development of new approaches for drug discovery is essential for accelerating the identification of effective treatments. In this work, we have proposed a machine learning-based method for drug testing, comparing two main strategies: classical machine learning models based on extracted features and deep learning models that directly analyze video sequences. In the case of traditional machine learning, we evaluated various classifiers, including Logistic Regression, Random Forest and XGBoost. All of them achieved similar results, although Random Forest showed slightly better performance, combining good accuracy with interpretability. On the other hand, deep learning models achieved promising results but were lower than those obtained with machine

learning (93% vs 96%). We identified two key limitations in the deep learning models: generalization issues and the lack of explicit information about speed and displacement. Our experiments showed that incorporating the position variation matrix significantly improved the model's performance, increasing the F1-score metric from 86% to 93%. To improve the results of the deep learning model, it would be necessary to filter out the cases of false detections and videos that do not appear centred on the worm, which can introduce noise in the training dataset. On the other hand, aggregation and edge cases appear infrequently in the dataset, so it would be necessary to increase the number of samples in the training. In the near future, approaches based on deep learning will be feasible, as continuous advancements are being made in skeletonisation and tracking models[23–27]. Once this is achievable, a new range of possibilities will open up by studying the new features learned by the model. These advancements in skeletonisation will also improve methods based on predefined features by increasing signal-to-noise ratio[27].

Since the results showed higher accuracy and explainability for the machine learning methods, we chose these models for the analysis of a published drug repurposing experiment. Our results show that the proposed approach offers advantages over the traditional statistical analysis used in O'Brien et al[15]. While O'Brien et al.[15] employed block permutation t-tests to detect phenotypic differences, our classifier-based approach provides a continuous recovery score, enabling a more quantitative assessment of drug effects. Traditional statistical methods rely on predefined features and binary significance testing (significant/not significant), whereas our model captures subtle, nonlinear phenotypic patterns and broader feature interactions. This advantage enhances sensitivity to potential treatments and mitigates biases associated with multiple comparison corrections. Despite some overlap with O'Brien et al.'s findings, our method identified distinct compounds, suggesting that machine learning offers a complementary alternative to standard statistical screening.

To illustrate this point, we trained a model specifically to distinguish between wild-type N2 worms and the *glr-4* mutant strain. Although no significant differences were detected in *glr-4* using statistical testing in the original dataset, our classifier achieved an F1-score of 77% on a held-out test set. This result demonstrates that meaningful phenotypic differences can be detected by machine learning even when standard statistical methods fail to find individual significant features, highlighting the added sensitivity and complementarity of our approach.

However, challenges remain. A key limitation of our study is the inability to validate our repurposing approach through an experiment where the truly effective compounds are already known. Verifying the utility of our identified hits would require human experimentation, which is not feasible at this stage. For this reason, the compounds identified by our method should be considered hypothesis-generating rather than confirmatory. Additionally, machine learning models, while powerful, are susceptible to overfitting and can lack interpretability. To address this, future work will explore explainability techniques such as SHAP (SHapley Additive Explanations)[28] to analyse feature contributions and refine model transparency.

In conclusion, our findings highlight the potential of machine learning to enhance automated phenotypic screening in *C. elegans*. By integrating machine learning with traditional statistical approaches, future research can further optimise drug discovery.

Data availability

The datasets (Tierpsy features, tracking data, and metadata) belong to the repository of another paper[15], and are available on Zenodo, https://doi.org/10.5281/zenodo.12547170. Any remaining information can be obtained from the corresponding author upon reasonable request.

Code availability

The code is available on GitHub at https://github.com/AntonioGarciaGarvi/MLDrugScreeningCelegans.

Received: 3 March 2025; Accepted: 3 July 2025
Published online: 18 July 2025

References

1. Kropp, P. A., Bauer, R., Zafra, I., Graham, C. & Golden, A. Caenorhabditis elegans for rare disease modeling and drug discovery: strategies and strengths. *Dis. Models Mech.* **14**, dmm049010. https://doi.org/10.1242/dmm.049010 (2021).
2. Nguengang Wakap, S. et al. Estimating cumulative point prevalence of rare diseases: analysis of the Orphanet database. *Eur. J. Hum. Genet.* **28**, 165–173. https://doi.org/10.1038/s41431-019-0508-0 (2020).
3. C. elegans Sequencing Consortium. Genome sequence of the nematode C. elegans: a platform for investigating biology. *Science (New York, N.Y.)***282**, 2012–2018, https://doi.org/10.1126/science.282.5396.2012 (1998).
4. Markaki, M. & Tavernarakis, N. Caenorhabditis elegans as a model system for human diseases. *Curr. Opin. Biotechnol.* **63**, 118–125. https://doi.org/10.1016/j.copbio.2019.12.011 (2020).
5. Alexander, A. G., Marfil, V. & Li, C. Use of Caenorhabditis elegans as a model to study Alzheimer's disease and other neurodegenerative diseases. *Front. Genet.* https://doi.org/10.3389/fgene.2014.00279 (2014).
6. Muñoz-Juan, A. et al. Caenorhabditis elegans RAC1/ced-10 mutants as a new animal model to study very early stages of Parkinson's disease. *Prog. Neurobiol.* **234**, 102572. https://doi.org/10.1016/j.pneurobio.2024.102572 (2024).
7. Patten, S. A. et al. Neuroleptics as therapeutic compounds stabilizing neuromuscular transmission in amyotrophic lateral sclerosis. *JCI Insight* https://doi.org/10.1172/jci.insight.97152 (2017).
8. Baek, J. H., Cosman, P., Feng, Z., Silver, J. & Schafer, W. R. Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *J. Neurosci. Methods* **118**, 9–21. https://doi.org/10.1016/S0165-0270(02)00117-6 (2002).
9. Geng, W., Cosman, P., Berry, C. C., Feng, Z. & Schafer, W. R. Automatic tracking, feature extraction and classification of C. elegans phenotypes. *IEEE Trans. Biomed. Eng.* **51**, 1811–1820. https://doi.org/10.1109/TBME.2004.831532 (2004).
10. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput. Biol.* **4**, e1000028. https://doi.org/10.1371/journal.pcbi.1000028 (2008).

11. Swierczek, N. A., Giles, A. C., Rankin, C. H. & Kerr, R. A. High-throughput behavioral analysis in C. elegans. *Nat. Methods* **8**, 592–598. https://doi.org/10.1038/nmeth.1625 (2011).
12. Yemini, E., Jucikas, T., Grundy, L. J., Brown, A. E. X. & Schafer, W. R. A database of Caenorhabditis elegans behavioral phenotypes. *Nat. Methods* **10**, 877–879. https://doi.org/10.1038/nmeth.2560 (2013).
13. Javer, A., Ripoll-Sánchez, L. & Brown, A. E. Powerful and interpretable behavioural features for quantitative phenotyping of Caenorhabditis elegans. *Philoso. Trans. R. Soc. B: Biol. Sci.* **373**, 20170375. https://doi.org/10.1098/rstb.2017.0375 (2018).
14. Barlow, I. L. et al. Megapixel camera arrays enable high-resolution animal tracking in multiwell plates. *Commun. Biol.* **5**, 1–13. https://doi.org/10.1038/s42003-022-03206-1 (2022).
15. O'Brien, T. J., Barlow, I. L., Feriani, L. & Brown, A. E. High-throughput tracking enables systematic phenotyping and drug repurposing in C. elegans disease models. *Elife* **12**, RP92491. https://doi.org/10.7554/eLife.92491 (2025).
16. García Garví, A., Layana Castro, P. E., Escobar-Benavides, S. & Sánchez-Salmerón, A.-J. Analysis of the Effect of Spatial and Temporal Resolution for the Classification of Caenorhabditis Elegans Movement Patterns Using Artificial Neural Networks. *SSRN Scholarly Paper***2024**, https://doi.org/10.2139/ssrn.4877386 (2024).
17. Javer, A. et al. An open-source platform for analyzing and sharing worm-behavior data. *Nat. Methods* **15**, 645–646. https://doi.org/10.1038/s41592-018-0112-1 (2018).
18. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778, https://doi.org/10.1109/CVPR.2016.90 (2016).
19. Deng, J. *et al.* Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255, https://doi.org/10.1109/CVPR.2009.5206848 (2009).
20. Dosovitskiy, A. *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. https://doi.org/10.48550/arXiv.2010.11929 (2021).
21. Pedregosa, F. et al. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
22. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems***32** (2019).
23. Hebert, L., Ahamed, T., Costa, A. C., O'Shaughnessy, L. & Stephens, G. J. Wormpose: Image synthesis and convolutional networks for pose estimation in c. elegans. *PLoS Comput. Biol.* **17**, e1008914. https://doi.org/10.1371/journal.pcbi.1008914 (2021).
24. Layana Castro, P. E., García Garví, A., Navarro Moya, F. & Sánchez-Salmerón, A.-J. Skeletonizing caenorhabditis elegans based on u-net architectures trained with a multi-worm low-resolution synthetic dataset. *Int. J. Comput. Vision* **131**, 2408–2424. https://doi.org/10.1007/s11263-023-01818-6 (2023).
25. Banerjee, S. C., Khan, K. A. & Sharma, R. Deep-worm-tracker: Deep learning methods for accurate detection and tracking for behavioral studies in C. elegans. *Appl. Anim. Behav. Sci.* **266**, 106024. https://doi.org/10.1016/j.applanim.2023.106024 (2023).
26. Alonso, A. & Kirkegaard, J. B. Fast detection of slender bodies in high density microscopy data. *Commun. Biol.* **6**, 1–12. https://doi.org/10.1038/s42003-023-05098-1 (2023).
27. Weheliye, W. H., Rodriguez, J., Feriani, L., Javer, A. & Brown, A. E. *An improved neural network model enables worm tracking in challenging conditions and increases signal-to-noise ratio in phenotypic screens* https://doi.org/10.1101/2024.12.20.629717 (2024).
28. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 4768–4777 (Curran Associates Inc., Red Hook, NY, USA, 2017).

Acknowledgements

We would like to express our gratitude to the Behavioral Phenomics group at the MRC Laboratory of Medical Sciences (LMS) for providing the datasets and for their valuable discussions. This study was supported by Ministerio de Universidades (Spain) under grant FPU20/02639 and EST24/00169. The authors thank the EU-FEDER Comunitat Valenciana 2014-2020 grant IDIFEDER/2018/025.

Author contributions

A.G.: Conceptualization, Methodology, Software, Data curation, Writing – original draft. A.S.: Conceptualization, Resources, Writing – review and editing. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-10370-x.

Correspondence and requests for materials should be addressed to A.-J.S.-S.