Deviation Error: assessing predictions for replicate measurements in genomics and beyond

Husam Abdulnabi and J. Timothy Westwood

Department of Cell and Systems Biology, University of Toronto Mississauga, Canada

Correspondence: husam.abdulnabi@mail.utoronto.com

Abstract

A quantitative measurement can have variation, referred to here as measurement variation, which is a probability distribution. Machine Learning models typically produce a prediction corresponding to the mode of the measurement variation. The Deviation Error is a novel metric, described here, to assess predictions that accounts for measurement variation.

Measurement variations in genomics data were explored. Towards a general prescription for modelling genomics measurements, different loss functions were used to fit models on synthetically generated data that mimics genomics measurements. Synthetically generated data offers the ability to know the true underlying value and to control the forms and amounts of noise injected at different stages of data processing. Different datasets were generated with varying levels of noise. Of the loss functions tried, only models fit with the Deviation Error performed as well if not better on any of the combinations of the metrics and datasets.

1 Introduction

A measurement of a quantity can vary due to the variability of the measurement methods and the nature of the quantity itself (e.g. biological phenomena). The variation of a measurement, referred to here as measurement variation, is described by a probability distribution and is estimated by repeating the measurement, referred to here as repeat or replicate measurements.

The probability distribution for a discrete quantity, known as a probability mass function (PMF), describes the probability of observing a specific discrete quantity from real measurements. Conversely, the probability distribution for a continuous quantity, known as a probability density function (PDF), describes the relative likelihood of observing a specific continuous quantity from real measurements. A PDF can be used to determine the probability of observing a specific range of continuous values through integration. Notably, a probability distribution can be used to determine the probability of a value occurring relative to another (**Sup. Eq. 1**).

In the simplest case, Machine Learning (ML) models produce a prediction corresponding to a *known* value. For measurements with measurement variation, the known value is the mode of the measurement variation. The accuracy of a prediction is typically assessed as the absolute or the squared difference of it and its corresponding known value, called the Absolute Error (**Eq. 1**) or the Squared Error (**Eq. 2**). A set of predictions is typically assessed as the average of the Absolute Errors or Squared Errors, called the Mean Absolute Error (also known as the L1 error, **Eq. 3**) or Mean Squared Error (**Eq. 4**), respectively.

Eq. 1. Absolute Error

$$AE(\hat{y}, y) = |\hat{y} - \mathrm{y}|$$

where:

$y =$ actual value

$\hat{y} =$ predicted value

Eq. 2. Squared Error

$$SE(\hat{y}, y) = (\hat{y} - \mathrm{y})^2$$

Eq. 3. Mean Absolute Error

$$MAE(\hat{Y}, Y) = \frac{1}{n}\sum_{i}^{n} AE(\hat{y}, y)$$

where:

$\hat{Y} = \{\widehat{y_1}, \widehat{y_2}, \dots, \widehat{y_n}\}$ set of all predicted values

$Y = \{y_1, y_2, \dots, y_n\}$ set of all actual values

Eq. 4. Mean Squared Error

$$MSE(\hat{Y}, Y) = \frac{1}{n}\sum_{i}^{n} SE(\hat{y}, y)$$

For quantitative measurements with variation, a corresponding prediction has a probability of occurring in real measurements. The prediction error should account for the probability of the prediction occurring. For two predictions corresponding to two different measurements with different measurement variations, the prediction that is more probable to occur within their corresponding probability distribution should score better than the prediction that is less probable, even when they both have the same absolute prediction error.

Notably, ML models can be used to predict the measurement variation by predicting the parameters of an assumed distribution, known as probabilistic learning (Murphy, 2022). For example, for measurement variation that is explained by a *t*-distribution, the model can predict the mean and sample standard deviation of the *t*-distribution. The prediction error, then, is determined by comparing the predicted and actual distributions. The objective of this study, however, is to assess a point prediction that corresponds to the actual value.

The measurement variation in genomics data, which is the primary impetus of this study, is explored in **Section 2**. A novel metric that effectively accounts for measurement variation, called the **Deviation Error**, is introduced in **Section 3**.

Functions that are used to assess the predictions can often be used as loss functions (also known as cost functions): functions that are used to adjust the learnable parameters of a ML model. Typically, the loss function is the same as the function that is used to assess the predictions. Different loss functions have been used in previous studies to model genomics data. Towards a general prescription for modelling genomics data, different loss functions are assessed using synthetic data that mimics genomics data in **Section 4**.

2 Measurement Variation in Genomics

Genomics is largely concerned with nucleotide sequences and the molecules that interact with them. For example, the strength of an interaction of a transcription factor to a specific region in the genome can vary. Measuring phenomena in genomics typically involves 4 steps: material preparation, sample preparation, DNA sequencing, and data processing.

86 Material preparation involves gathering biological material and fixing it. Fixing halts and
87 preserves molecular interactions, creating a snapshot of the molecular state in the biological
88 material.

89 Sample preparation involves isolating target nucleotide sequences (typically 25-250
90 nucleotides long) from the material prepared that is essentially a population of nucleotide
91 sequences, referred to here as base-level nucleotide sequences. If the target nucleotide sequences
92 are RNA, they are converted into DNA sequences. The DNA sequences gathered from
93 experiments, referred to here as experimental DNA, are identified by DNA sequencing.

94 There are different methods for DNA sequencing, but High-Throughput Sequencing
95 (HTS) is the current standard and thus the focus of this study. The identified DNA sequences
96 from HTS are known as sequencing reads, simply referred to as reads hereafter.

97 Most often, the experimental DNA derives from known nucleotide sequences, referred to
98 here as reference sequences, which are typically the genome of the biological material. Data
99 processing entails mapping the reads to the reference sequences and assigning each position of
100 the reference sequences a score based on the number of reads mapping to it. The scores are
101 known as read counts and are non-negative integers.

102 DNA sequencing and sample preparation involve sampling steps which results in
103 measurement variation. HTS identifies a sample of the DNA in the experimental DNA.
104 Repeating HTS on the same experimental DNA can yield different reads. Then, for a particular
105 position of a reference sequence, the number of reads mapping to it can vary, referred to here as
106 sequencing variation. The sequencing variation of a read count is best explained by a Negative
107 Binomial distribution (NBD) (Love et al., 2014; McCarthy et al., 2012). NBDs differ depending
108 on the expected number of read counts, exemplified in **Fig. 1. A**. To estimate the expected

numbers of the read counts, sequencing must be repeated on the same experimental DNA, referred to here as sequencing replicates.

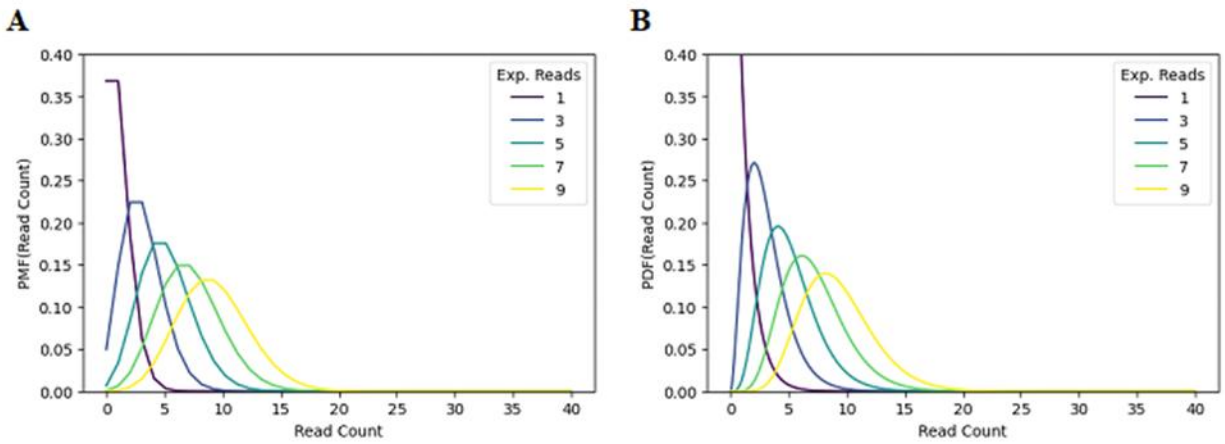


Fig. 1. The Gamma distribution is the continuous analog of the Negative Binomial distribution.

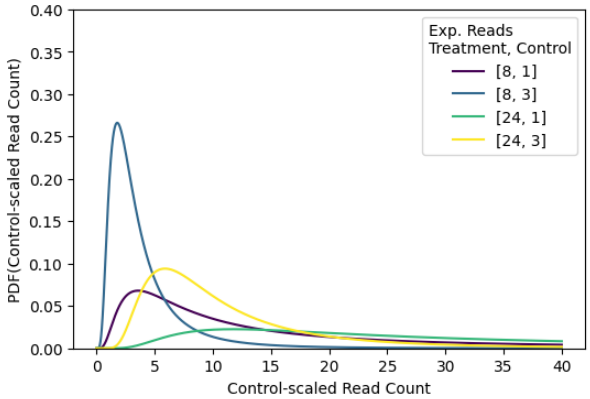
The Negative Binomial (**A**) and the Gamma (**B**) distributions with different expected read counts ("Exp. Reads"). The read counts for the (**B**) are [1e-10, 40].

The total number of reads produced from HTS can be roughly controlled. A greater number of reads yields better resolution than a lesser number of reads but is more expensive. Despite efforts to control the number of reads, the number of total reads for different sequencing replicates can vary. To make sequencing replicates comparable, the read counts of the sequencing replicates need to be scaled which is often based on the average total number of reads among the replicates.

Scaling often results in continuous quantities. Also, depending on the objective of the experiment and the number of reads used for HTS, the read counts may be smoothened by an averaging window, also yielding continuous values. The continuous analog of the NBD is the

127 Gamma distribution (GD, **Sup. Eq. 2**), exemplified in **Fig. 1. B**. Notably, the GD requires the
128 actual value be greater than 0 while read counts can be 0. To address this, a positive constant can
129 be added to all read counts or the read counts can be clipped to be some minimum value.

130 The read counts also contain two sources of bias, referred to here as sequence bias. First,
131 some DNA sequences are more preferentially sequenced than others (Amemiya et al., 2019).
132 Second, the number of base-level nucleotide sequences containing a particular position of the
133 reference sequences can vary. To account for the sequence bias, a sample of base-level
134 nucleotide sequences is sequenced, commonly known as an input control, referred to here as a
135 sequencing control. The sequencing control also has sequencing variation which too requires
136 replicates and is best modelled by a GD. The read counts of the target DNA are divided by the
137 corresponding read counts of the sequencing control. The ratio of two GDs is a Beta Prime
138 distribution (BPD) (**Sup. Eq. 3**, **Sup. Eq. 4**). The BPDs of different pairs of target and
139 sequencing control read counts (simply target-control pairs hereafter) are exemplified in **Fig. 2**.
140



141

142 **Fig. 2. Beta Prime distributions of different target-control pairs.**
143 Target and control expected read counts ("Exp. Reads") are noted as [target, control].
144 The control-scaled read counts are [1e-10, 40].

145

146 Sample preparation typically entails numerous sampling steps, resulting in measurement
147 variation in read counts that too may be explained by NBDs. Still, sample preparation also
148 contains other variation that may not be adequately explained by NBDs. Estimating the variation
149 in sample preparation requires repeating sample preparation from the same biological material,
150 referred to here as sample preparation replicates. To distinguish the variation from sequencing
151 and from sample preparation, each sample preparation replicate needs to have sequencing
152 replicates.

153 Furthermore, phenomena in genomics can vary, referred to here as phenomenal variation.
154 For example, a transcription factor may interact with less variation to one location in the genome
155 than to another. To estimate the phenomenal variation, material preparation needs to be repeated,
156 known as biological replicates. To distinguish the phenomenal variation from measurement
157 variation (which includes both variation from sequencing and sample preparation), each
158 biological replicate needs to have sample preparation replicates.

159 Due to costs and/or limited biological material, most genomic experiments consist of 2-3
160 biological replicates of the target and 1-3 replicates of the sequencing control that may be
161 biological, sample preparation, or sequencing replicates depending on the experiment. With
162 relatively little information, the biological variation cannot be reliably distinguished from the
163 sequencing and sample preparation variation. So, the replicates for the experiment and the
164 sequencing control are treated as sequencing replicates. As mentioned, the variations of
165 measurements of control-scaled sequencing replicates are best explained by BPDs. Thus, for
166 genomics experiments with these or similar replicates, the variations of measurements are best
167 explained by BPDs.

3 Deviation Error

The prediction error for measurements with measurement variation should account for measurement variation; it should account for the probability of observing the prediction in real measurements.

Under the assumption of homoscedastic, independent errors, minimizing the Mean Squared Error is equivalent to maximum-likelihood estimation with Gaussian noise (i.e. measurement variation explained by Normal distributions) **(Sup. Eq. 6**), while minimizing the Mean Absolute Error corresponds to maximum-likelihood estimation with Laplace (double-exponential) noise (Casella & Berger, 2002). Under those assumptions but satisfied in log-space, the Mean Squared Log Error (**Eq. 5**) can be used for log-scale Gaussian noise (**Sup. Eq. 7, Sup. Fig. 1**) and the Mean Absolute Error can be used for log-scale Laplace noise.

Eq. 5. Mean Squared Log Error

$$MSLE\left(\hat{Y}, Y, q\right) = \frac{1}{n}\sum_{i=1}^{n}(\log(\hat{y}_i + q) - \log(y_i + q))^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\log\left(\frac{\hat{y}_i + q}{y_i + q}\right)\right)^2$$

where:

$q =$ a constant to avoid log(0), $q \geq 0$

Another method that can be used to capture measurement variation in prediction assessment is the *z*-score or the *t*-score, yielding metrics such as the Mean Squared *z*-score Error (**Eq. 6**). Still, these methods assume that the measurement variation can be explained by Normal or *t*- distributions, respectively.

Eq. 6. Mean Squared *z*-score Error

$$MSZE(\hat{Y}, Y_\mu, Y_\sigma) = \frac{1}{n}\sum_i^n \left(\frac{\widehat{y_i} - \mu_i}{\sigma_i}\right)^2$$

where:

$Y_\mu = \{\mu_1, \mu_2, \dots, \mu_n\}$ set of all actual expected (mean) values

$Y_\sigma = \{\sigma_1, \sigma_2, \dots, \sigma_n\}$ set of all actual standard deviations

The methods mentioned have assumptions. If the measurement variations do not satisfy these assumptions, then these methods may no longer be sufficiently effective.

To directly accommodate any probability distribution whether homoscedastic or heteroscedastic in any scale (i.e. probability distribution agnostic), the Negative Log Likelihood (NLL) can be used (**Eq. 7**). The NLL derives from the likelihood which is used to assess prediction as the product of the probabilities of occurring since each prediction corresponds to an independent observation. To avoid arithmetic underflow, the log of the likelihood is used which is the sum of the log of the probabilities. Notably, the distribution of values in log scale are different than the distribution of values in normal scale, exemplified in **Fig. 3** following the example in **Fig. 2**. To accommodate different measurement variations without disrupting the probability distributions, the negative of the mean of the probabilities can be used, referred to here as the Negative Mean Probability (**Eq. 8**).

Eq. 7. Negative Log Likelihood

$$NLL(\hat{Y}; M) = -\sum_{i=1}^{n} \log(\max(P(\widehat{y_i};\ m_i), \varepsilon))$$

where:

M = set of all admissible parameter configurations

m = an arbitrary set of function parameters, $m \in M$

$P(x; m)$ = probability distribution function

ε = a small constant to avoid log(0), $\varepsilon \geq 0$

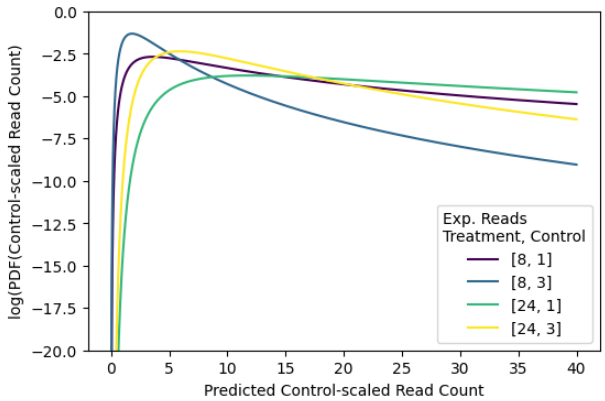


Fig. 3. Log probabilities have different shapes than the original probabilities. Following **Fig. 2**, log probabilities of the predicted control-scaled read counts. The y-axis is clipped to [0, -20]. See **Fig. 2** for additional descriptions.

Eq. 8. Negative Mean Probability

$$NMP(\hat{Y}; M) = -\frac{1}{n}\sum_{i}^{n} P(\hat{y}_i;\ m_i)$$

Importantly, different probability distributions can have different maximums, exemplified previously in **Fig. 2**. Accurately predicting the mode of one measurement variation may not reward/penalize the same as predicting the mode of another. Following the example in **Fig. 2**,

229 predicting the mode of a target-control pair of [8,3] yields less error than predicting the mode of
230 a target-control pair of [8,1]. This extends to log probabilities which is used in NLL.
231 Furthermore, common probability distributions in genomics and outside genomics
232 include unimodal distributions which consist of a single peak and decreasing probability away
233 from the mode (e.g. Gamma, BPD, and Laplace distributions). These distributions can plateau
234 where differences between predicted values are similar. Following the example in **Fig. 2**,
235 predictions that are greater than 20 for a target-control pair of [8,3] have similar probabilities of
236 occurring. Log probabilities (as used in NLL) of these unimodal distributions do not plateau but
237 as mentioned, they do not preserve the original probability distribution.
238 To comparably assess predictions corresponding to different probability distributions
239 with different maximums and to serve probability distributions that plateau while better
240 preserving the shape of probability distributions, the **Deviation Error** is developed. The
241 Deviation Error is the product of the complement of the maximum-scaled probability distribution
242 and the Absolute Error (**Eq. 9**). Notably, the values of a maximum-scaled probability distribution
243 have a range of [0, 1]. To make an error function where lower values are preferred and not
244 needing to use log which alters the shape of the distribution, the complement is used. For
245 assessing a set of predictions, the Mean Deviation Error can be used (**Eq. 10**).
246
247 **Eq. 9. Deviation Error**
248 $$DE(\hat{y}, y; m) = \left(1 - \frac{P(\hat{y}; m)}{\max(P(m))}\right) * AE(\hat{y}, y)$$
249
250 **Eq. 10. Mean Deviation Error**

$$MDE(\hat{Y}, Y; M) = -\frac{1}{n}\sum_{i}^{n} DE(\hat{y}_i, y_i; m_i)$$

Following the example in **Fig. 2**, the Absolute Error, the NLL, and the Deviation Error for different target-control pairs are compared in **Fig. 4**. The shapes of the Absolute Errors for all pairs are the same. The shapes of the NLL for different pairs differ but it does not preserve the original probability distribution and different target-control pairs have different minimums: a correct prediction for one target-control pair is rewarded/penalized differently than another. Conversely, the shapes of the Deviation Errors for the different pairs are different but better preserves the original distribution, and correct predictions are rewarded/penalized the same.

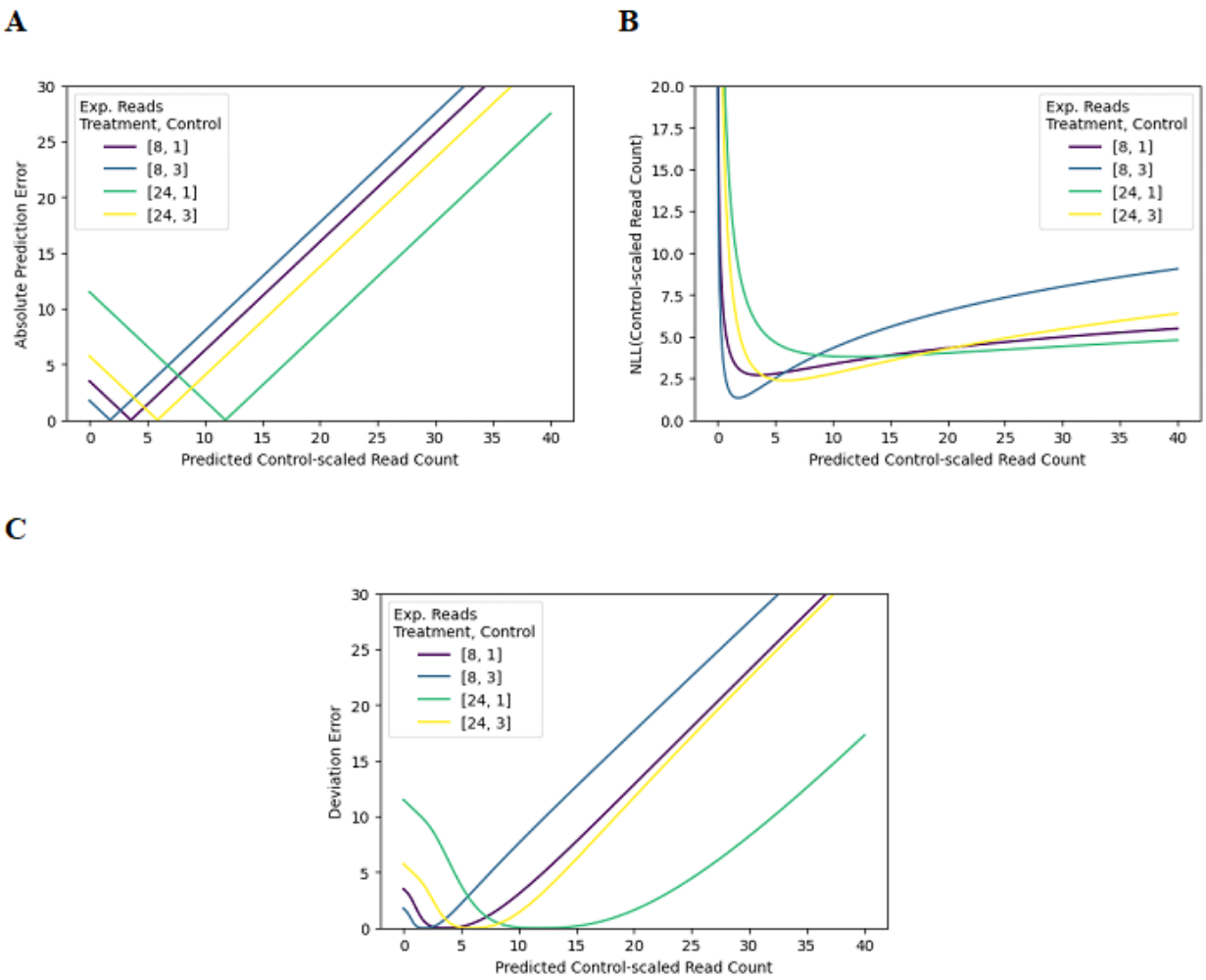


Fig. 4. The Deviation Error preserves distributional shape while applying comparable penalties better than the Absolute Error and the Negative Log Likelihood.

The (**A**) Absolute Error, (**B**) NLL, and the (**C**) Deviation Error of predicted control-scaled read count for target-control pairs.

See **Fig. 2** for additional descriptions.

3.1 Further Connections with Other Metrics

The Deviation Error can be interpreted as an absolute error weighted by the likelihood of the prediction occurring in real measurements. Several alternative probability-based weighting

strategies have been proposed. In confidence-weighted regression, each residual is scaled by the inverse of its standard deviation (1/σ), reflecting the principle of heteroscedastic uncertainty weighting. Probabilistic regression methods extend this idea by predicting a full noise distribution alongside the mean, whether through scaling the MSE by a predicted variance (Nix & Weigend, 1994) or via Bayesian deep learning approaches that jointly estimate mean and variance (Kendall & Gal, 2017).

A different family of scoring rules derives from the Tsallis entropy (Tsallis, 1988). The Tsallis scoring rule, or q-score, generalizes the log-likelihood by introducing a parameter q (**Eq. 11**). As q approaches 1, the rule penalizes low-probability predictions more harshly, whereas larger q values temper this effect. By setting $q = 2$, the Tsallis score reduces to $\left(1 - p(y)\right)$ which resembles the core weighting term of the Deviation Error.

Eq. 11. Tsallis' q-score

$$S_q(p, y) = \frac{1 - [p(y)]^{q-1}}{q - 1}, q > 0, q \neq 1$$

Focal-style losses, originally developed as Focal Loss for classification to address class imbalance (Lin et al., 2017), similarly modulate scores based on the probability of observing predictions in real measurements (**Eq. 12**). The weighting term $(1 - p_t)^\gamma$ is based on the model's prediction after softmax which is a probability. This extends to $\left(1 - p(y)\right)$ for regression which resembles the core weighting term of the Deviation Error.

Eq. 12 Focal Loss

$$FL(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t)$$

where:

p_t = model's estimated probability

γ = focusing parameter

α_t = class imbalance parameter

The weighting term from the reduction of Tsallis' q-score and Focal-style losses can be defined as the complement probability. Notably, while PMFs have a lower limit of 0 and a upper limit of 1, PDFs have a lower limit of 0 but no upper limit. We define weighting the Absolute Error by the complement probability as the Complement-weighted Error (CWE) (**Eq. 13**). For assessing a set of predictions, the Mean Complement-weighted Error (MCWE) can be used (**Eq. 14**).

Eq. 13. Complement-weighted Error

$$CWE(\hat{y}, y; m) = \left(1 - P(\hat{y}; m)\right) * V(\hat{y}, y)$$

Eq. 14. Mean Complement-weighted Error

$$MCWE\left(\hat{Y}, Y; M\right) = \frac{1}{n}\sum_{i}^{n} CWE(\hat{y}, y; m)$$

Furthermore, Robust M-estimators such as density-power divergence weighting (Basu, 1998), Huber loss (Huber, 1964), and Tukey's Biweight (Beaton & Tukey, 1974) offer alternative means to mitigate the influence of outliers. Density-power divergence weights residuals by the probability density of the observation, while Huber and Tukey functions apply piecewise thresholds to cap the impact of large errors. In contrast, the Deviation Error intensifies

317 penalties for large residuals that occur in low-probability regions of a unimodal distribution
318 compared to the NLL, making it an "inverse" M-estimator that emphasizes unlikely deviations
319 rather than suppressing them.
320

321 4 Synthetic Genomics Data

322 Real world measurements are estimations of a *true* quantity. To improve estimates,
323 measurements are replicated, yielding a probability distribution. The objective in modelling is to
324 produce predictions that minimize the difference between it and the true quantity. As mentioned,
325 the difference is assessed by the Absolute Error or the Squared Error – the choice of which is
326 ultimately subjective.
327 The true quantity cannot be determined in the real world and accurately estimating the
328 amount and form of noise in each measurement is not trivial. Synthetic experiments offer a way
329 around this limitation: the generation of synthetic data provides the true quantity and allows the
330 control of noise at various steps.
331 The following case study synthetically generates data and compares the use of different
332 metrics as loss functions towards a prescription for modelling measurements with measurement
333 variation. More specifically, the following case study generates data that mimics genomics data
334 towards a prescription for modelling genomics data.
335 As explored, genomics data is often a result of HTS and involves target and control
336 samples. The read counts of the target sample are scaled by the read counts of the control
337 sample. Genomics data often has 3 replicates which primarily serve to capture variation from
338 HTS which typically follow Gamma distributions. To gauge the variation from earlier steps in

339 the experiment (e.g. biological phenomena, sample preparation, etc.) more and different
340 replicates are needed but are seldom done.

341 A randomly initialized NN is used to generate true target-control read pairs. The true
342 target-control read count is determined as the ratio of the true target reads and the true control
343 reads. To reflect variation from earlier steps different amounts of log-scale Gaussian noise is
344 injected to the true target-control read pairs, producing *pre-sequencing* target-control read pairs.
345 We do not claim that variation prior to sequencing is strictly log-scale Gaussian as they surely
346 follow more complex distributions especially those derived from biological phenomena. We
347 adopt this approximation for its simplicity and for it yielding non-negative values (as sequencing
348 data is non-negative) towards data that better resembles real genomic assays.

349 To reflect sequencing variation and typical experiments, three gamma samples are
350 produced for each read count and averaged, producing *post-sequencing* (post-seq) read counts.
351 The post-seq target-control read count for a target-control pair is determined as the mode of a
352 Beta Prime distribution parameterized by the post-seq target and control read counts. The post-
353 seq target-control read counts are the values that the model predicts for in model fitting.

354 For ML genomics data, several metrics have been used as loss functions. The MSE (e.g.
355 Alipanahi et al., 2015) is one of the simplest and is widely used in ML in general and we use this
356 as the standard loss function that we compare to.

357 Variation from Gamma and Beta Prime distributions exhibit variation that increases with
358 the magnitude of the values (exemplified in **Sup. Fig. 2**), resembling log-scale Gaussian noise
359 (**Sup. Fig. 1**). As mentioned, log-scale Gaussian noise can be modelled using MSLE. When
360 values can be 0, the MSLE requires a positive constant to be added to avoid log(0). A constant of
361 1 has been used previously (e.g. Avsec et al., 2021), referred to here as MSLE-1. A small

constant can also be used, referred to here as MSLE-eps. Another similar approach that accounts variation that increases with magnitude is the Mean Poisson Negative Log Likelihood (P-NLL, **Eq. 15**) (e.g. Kelley, 2020).

Eq. 15. Mean Poisson Negative Log Likelihood

$$PNLL(\hat{Y}, Y, q) = \frac{1}{n}\sum_{i=1}^{n} \hat{y_i} - y_i * (\log(\hat{y_i} + q))$$

As explored, the NMP, NLL, the MDE, and the MCWE can all be used to assess predictions for measurements with measurement variation and thus are also tried. Here, these functions will assume the measurement variation is a Beta Prime distribution.

The predictions on the testing set are compared to the corresponding post-seq control-scaled read counts, assessed by the NLL and the MDE. The predictions on the testing set are also compared to the corresponding true control-scaled read counts – an ability unique to synthetic data – assessed by MAE and MSE. Comparing predictions to the measurement mimicking real-world experiments and to the true values with multiple metrics provides a more objective and complete comparison of the loss functions towards a general prescription for modelling genomics data.

4.1 Methods

All code is available at https://github.com/Husam94/DevLoss.

4.1.1 Neural Networks

Inspired by Steininger et al., 2021, all NNs used have 3 Dense layers each with 10 nodes and ReLU activation functions between the Dense layers. Batch normalization layers were added between the Dense layers as they are found in most of the current NN architectures. For data generation, a NN with two outputs is used to produce target-control pairs, referred to here as the Data NN. For modelling, a NN with one output is used to predict the modes of the BPDs of the target-control pairs, referred to here as the Model NN. The Model NN has a ReLU activation function at the end so that values have a minimum of 0.

4.1.2 Data Generation

4.1.2.1 Initial Data Generation

A 10-feature input was generated by sampling uniform distributions with a range [-1, 1]. A total of 1,000,000 10-feature inputs were generated. A Data NN was randomly initialized by Kaiming Initialization and produced corresponding 2-feature outputs (representing target and control values) from the inputs. The pairs of input-output are referred to as original data. This procedure was repeated 20 times, generating 20 sets of original data from 20 different initializations of Data NNs, referred to as synthetic repeats. The outputs of the synthetic repeats were visually inspected for bell-shaped distributions. Of the synthetic repeats, 5 were chosen that best resemble bell-shaped distributions for the target and control values.

4.1.2.2 Selection

Observations of 2-feature outputs with values inside 1 standard deviation for each feature were kept; the others were trimmed. It was preferred that the observations have a diverse

407 combination of target-control values. The target and control values were separately binned to 10
408 equally spaced bins. The observations were binned as a combination of their target and control
409 bins, producing (10*10) 100 bins.

410 For each dataset, 20,000 observations were desired based on computing budgets. The
411 observations were iteratively selected without replacement from each of the 100 bins (in random
412 order) until the desired number of observations were obtained.

413 Based on personal experience with real genomics data, the target and the control values
414 were scaled to ranges [5, 200] and [5, 20], respectively, referred to as true read counts. The true
415 control-scaled values are determined as the ratio of the true target and control read counts.

416

417 **4.1.2.3 Log-scale Gaussian Noise**

418 Log-scale Gaussian Noise was added to the true read counts (**Eq. 16**), producing noised
419 read counts, referred to as pre-sequencing (pre-seq) read counts. Noise was initially generated
420 using a truncated normal distribution with a mean of 0, a standard deviation of 0.5, and bounds
421 of ±1 standard deviations. Different amounts of noise were controlled by a parameter g. "None",
422 "Low", and "High" noise were produced with g of 0, 0.5, and 1.0, respectively. Notably, "None"
423 is devoid of log-scale Gaussian noise. The pre-seq control-scaled read counts are determined as
424 the ratio of the pre-seq target and control read counts.

425

426 **Eq. 16. Log-scale Gaussian noise with truncated normal distribution**

427 $$y_{noised} = y_{true} * e^{g*\mathrm{TN}(0,0.5,1)}$$

428 where:

429 y_{true} = true output values

g = level of noise

$\mathrm{TN}(\mu, \sigma^2, b)$ = truncated normal distribution with a mean (μ), standard deviation (σ) and bounds of $\pm b$

y_{noised} = noised output values

4.1.2.4 Gamma Noise

For each of the pre-seq target and control read counts, three samples were drawn from respective GDs with alphas of the true read count and a scale of 1, mimicking sequencing replicates. The samples were then averaged, referred to as post-sequencing (post-seq) read counts. The post-seq control-scaled read counts were determined as the mode of the BPD with alpha and beta as the post-seq target and control read counts. Examples of comparing true, pre-seq, and post-seq control-scaled read counts for different amounts of log-scale Gaussian noise are provided in **Fig. 5**. For each synthetic repeat, the selection, injections of log-scale gaussian and gamma noise, were repeated 3 times, referred to as data processing repeats.

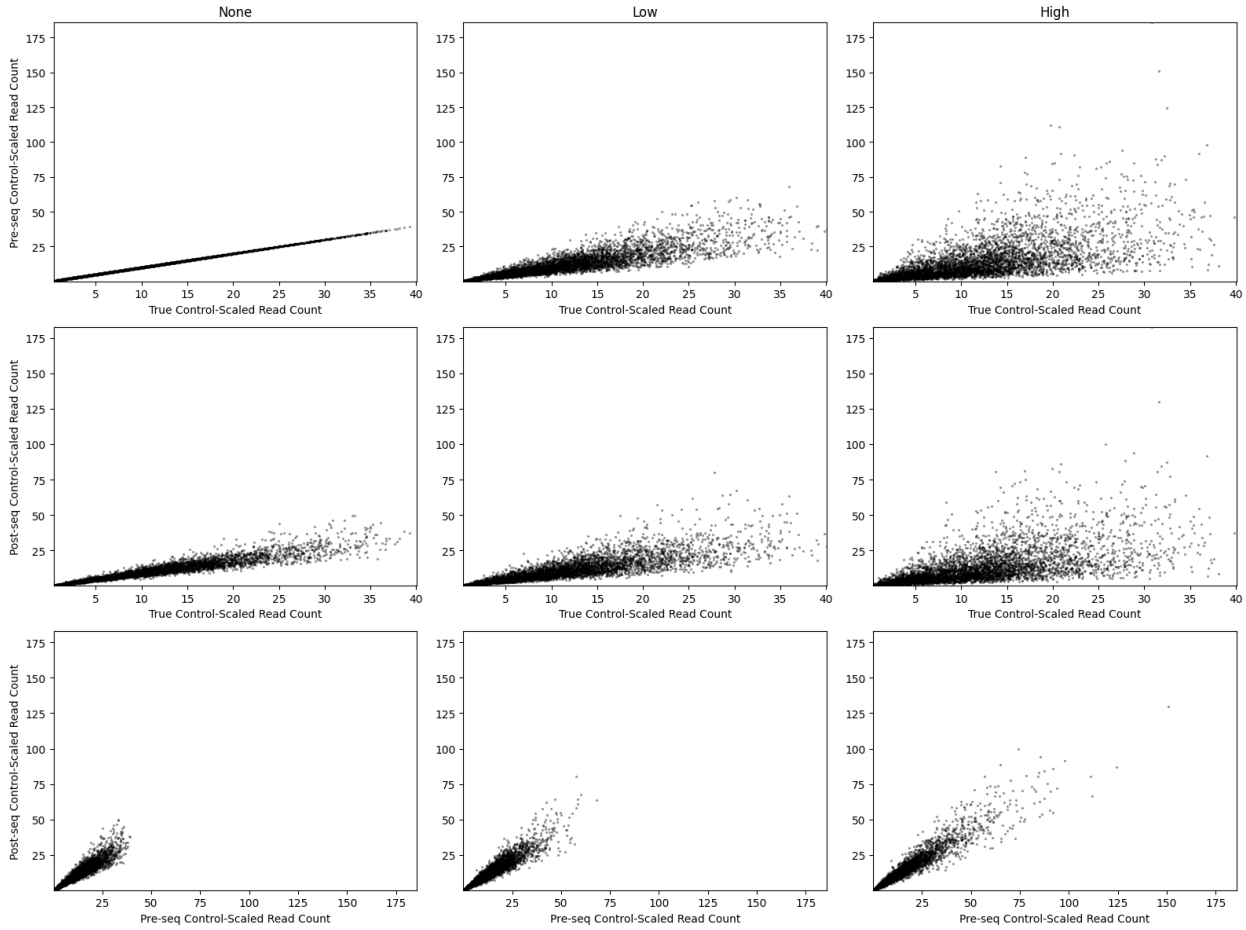


Fig. 5. Comparing control-scaled read counts at different stages of data generation. Data is from one of the data processing repeats. Columns are the levels of noise and each row compares either (first) true and pre-seq control-scaled read counts, (second) true and post-seq control-scaled read counts, and (third) pre-seq and post-seq control-scaled read counts.

4.1.2.5 Data Splitting

For each dataset, the observations were binned into 10 equally spaced bins by the post-seq control-scaled read counts. The data was split into training, stoppage, and testing sets with proportions of 0.6, 0.2, 0.2, respectively, by Stratified split (see **Suppl. Note 9.3.1**). For each

455 synthetic repeat and data processing repeat, data splitting was repeated 3 times, referred to as
456 data split repeats.
457

458 ### 4.1.3 Model Fitting

459 All modelling used the Model NN to predict the post-seq control-scaled read counts, a
460 batch size of 512, a stoppage patience of 10 epochs (see **Suppl. Note 9.3.1**), and the Adam
461 Optimizer with a learning rate of 0.0124 (Kingma & Ba, 2014). The learning rate used for the
462 experiments was determined as the learning rate that best reduces the MSE on the validation set
463 with MSE as the loss function and a budget of 200 epochs (**Sup. Table 1**). The stoppage metric
464 was the same as the loss function (e.g. the MSE was used as the stoppage metric when MSE was
465 used as the loss function). Unless otherwise specified, a constant value of 1e-10 is used for all
466 loss functions to avoid log(0) or dividing by 0.
467

468 ### 4.1.4 Scoring Experimental Units

469

470 #### 4.1.4.1 Ensemble Predictions

471 The predictions on the testing set of a model were primarily scored against the post-seq
472 values and by the loss function it was fit with.
473 The learnable parameters of NNs are initialized randomly and are fit using stochastic
474 methods which can result in substantially varying model performances. Considering computing
475 budgets, each experimental unit was repeated 15 times, referred to as model repeats. Random
476 initializations can result in models that produce predictions that are only below 0 before the final
477 ReLU activation and thus produce predictions that are all 0 after ReLU activation which disables

478 model learning, referred to as a dud initialization. To avoid dud initializations, a model with a
479 dud initialization is re-initialized up to 15 times.

480 Commonly, a single model out of a set of model repeats that best performs on the
481 stoppage set was used for further assessment. This selection criteria, however, lacks
482 reproducibility as a different set of model repeats can yield a substantially different best
483 performing model. Instead of using a single best performing model, the best 3 models were used.
484 These were determined as the best scoring models on the stoppage set assessed by the loss
485 function they were fit with and against the post-seq control-scaled read counts. The predictions
486 of the best 3 models were averaged, akin to an average ensemble model, producing ensemble
487 predictions.

488 For later statistical assessment, different sets of model repeats were simulated by
489 bootstrapping the existing model repeats (Efron & Tibshirani, 1986). Based on computing
490 budgets, 50 bootstrap sets of model repeats were sampled for each set of model repeats, and the
491 ensemble predictions were determined for each bootstrap set, producing 50 bootstrap ensemble
492 predictions.

493 The bootstrap ensemble predictions were scored by comparing to the true control-scaled
494 read counts with MSE and MAE and comparing to the post-seq control-scaled read counts with
495 NLL and MDE.

496

497 **4.1.4.2 Comparisons of Experimental Units**

498 A subject experimental unit was compared to a corresponding reference experimental
499 unit which was the experimental unit that matched the evaluating metric. For example, when

comparing the MSE of the ensemble predictions and the true control-scaled read counts, the models fit with the MSE were used as the reference experimental unit.

A subject was compared to the reference by pairwise comparisons of their bootstrap scores, resulting in (50*50) 2,500 comparisons which simulates repeating the modelling of the two experimental units 2,500 times. Each pair of bootstrap scores were compared by percent relative change, referred to as the pairwise bootstrap score. Negative pairwise bootstrap scores indicate the subject has a lower score than the reference which is preferred. For each noise level, the pairwise bootstrap scores of synthetic repeats, data processing repeats, and data splitting repeats were pooled, totaling (5*3*3*2,500) 112,500 pairwise bootstrap scores.

The mean and the standard error of the pooled pairwise bootstrap scores were determined for each dataset and for each experimental unit. For statistical assessment, the 5th (P_5) and the 95th (P_{95}) percentiles of the pairwise bootstrap scores were determined, simulating a one-sided statistical significance level of 0.05 and a two-sided statistical significance level of 0.1 (Efron & Tibshirani, 1986). If $0 > P_{95}$, percentile, the subject model is significantly better than the reference model. If $0 < P_5$ percentile, the subject model is significantly worse than the reference model. If $P_5 < 0 < P_{95}$, the subject model is not significantly different from the reference model.

4.2 Results and Discussion

Models were fit with different loss functions on different sets of data that varied by the amounts of log-scale Gaussian noise. For each dataset, the predictions of the sets of models fit with different loss functions were evaluated by comparing with the true control-scaled read counts with MSE or MAE and by comparing with the post-seq control-scaled read counts with NLL and MDE. (**Table 1**).

Table 1. Models fit with the Deviation Error perform as well if not better compared to reference models based on multiple metrics comparing predictions against true and post-seq control-scaled read counts.

Mean pairwise bootstrap scores comparing fitting with different loss functions to fitting with a reference loss function which is the same as the evaluating metric (e.g. the reference loss function is the NLL when evaluating with the NLL). The predictions of the models were compared to the true control-scaled read count with MSE and MAE and compared to the post-seq control-scaled read count with NLL and MDE.

The "Shuffled MSE" refers to models fit with MSE as the loss function but with the outputs in the training and validation sets shuffled, serving as a negative control.

See **Sup. Table 2** for full bootstrap statistics.

Negative values indicate that models fit using the subject configuration performed better than models fit using the reference configuration and vice-versa. Bolded values indicate that the models fit using the subject configuration are significantly better or worse than models fit using the reference configuration.

"Noise" refers to the level of additional log-scale Gaussian noise injected. Using the Deviation Error in a loss function (MDE) is shaded in purple.

Target	**True control-scaled read count**						**Post-seq control-scaled read count**					
Metric	**MSE**			**MAE**			**NLL**			**MDE**		
Noise	**None**	**Low**	**High**	**None**	**Low**	**High**	**None**	**Low**	**High**	**None**	**Low**	**High**
Loss Functions												
MSE	0.64	0.82	0.52	-8.41	**-19.40**	-4.86	2.05	3.96	**8.09**	10.64	4.54	**3.79**
Shuffled MSE	**1472.11**	**1453.62**	**927.25**	**291.37**	**247.70**	**223.68**	**98.83**	**82.28**	**56.53**	**749.88**	**189.56**	**58.42**

MAE	27.26	**61.24**	30.08	0.15	0.17	0.17	-0.35	0.38	0.99	3.91	2.48	1.03
MSLE-eps	**132.14**	**142.39**	**75.73**	**28.53**	17.33	11.34	3.37	2.15	1.37	**47.21**	**10.29**	2.39
MSLE-1	31.47	**48.13**	12.43	-2.21	-8.19	-11.11	-2.16	-1.96	-1.02	0.03	-0.64	-0.36
P-NLL	12.68	4.16	-1.09	-7.10	**-21.05**	-6.72	0.76	1.77	**5.88**	10.29	2.69	**2.80**
NLL	**88.05**	**101.56**	32.06	15.65	7.00	-3.19	0.08	0.08	0.01	19.84	4.94	0.69
NMP	**298.42**	**543.76**	**886.35**	**65.97**	**101.26**	**205.48**	6.21	**15.16**	**50.66**	**82.47**	**48.56**	**41.81**
MDE	14.52	13.30	-18.30	-5.66	**-17.65**	**-20.50**	-0.98	0.02	1.45	0.25	0.02	0.00
MCWE	25.61	**54.54**	27.48	-0.82	-2.52	-1.12	-0.26	0.14	0.81	4.17	1.86	0.90

4.2.1 Comparing Predictions to the True Control-scaled Read Counts

Using MSE as the evaluating metric and models fit with MSE as the reference experimental unit, none of the other loss functions produced significantly better models. Still, there were loss functions that produced significantly worse models for different levels of noise: MAE, MSLE-eps, MSLE-1, NLL, NMP, and MCWE (**Table 1**).

Using MAE as the evaluating metric and models fit with MAE as the reference experimental unit, only the MSE, P-NLL, and the MDE produced significantly better models for datasets with low levels of noise. Only the MDE produced significantly better models for datasets with high levels of noise. Conversely, the MSLE-eps and the NMP produced significantly worse models for different levels of noise.

Of the loss functions, only the P-NLL and the MDE were significantly better than the reference loss function on some of the levels of noise while not being significantly worse for any of the levels of noise.

4.2.2 Comparing Predictions to the Post-seq Control-scaled Read Counts

Using NLL as the evaluating metric and models fit with NLL as the reference experimental unit, none of the other loss functions produced significantly better models for any of the levels of noise. Still, there were loss functions that produced significantly worse models

for different levels of noise: P-NLL (for high noise) and the NMP (for low and high noise) (**Table 1**).

Using MDE as the evaluating metric and models fit with MDE as the reference experimental unit, none of the other loss functions produced significantly better models for any of the levels of noise. Still, there were loss functions that produced significantly worse models for different levels of noise: MSLE-eps (for no and low noise), P-NLL (for high noise) and the NMP (for all noise).

Of all the loss functions tried, among the different combinations of control-scaled read counts and metrics, only models fit with the Mean Deviation Error were as good (i.e. not worse) if not better than reference models. Therefore, at least for data with measurement variation that mimics genomics data, the Mean Deviation Error is the preferred loss function among the loss functions tried.

Further exploration into the differing performances of the different loss functions on different levels of noise is worthwhile but is beyond the scope of this paper. It is presumed that the differing performances are due to how the loss functions shape the NN's gradients for model fitting (Li et al., 2018). Importantly, all models were fit using the Adam optimizer with the same learning rate that was selected based on a separate trial trying different learning rates with fitting with the MSE as the loss function and evaluating with the MSE on the post-seq control-scaled read counts (**Sup. Table 1**). Each loss function may prefer a different optimizer and/or learning rate (or other tuneable parameters) and so optimization of the optimizer for each loss function is warranted but was not pursued here (Barati et al., 2025).

5 Conclusion

The Deviation Error is a novel metric that effectively accounts for both absolute prediction error and measurement variation. It is recommended to be used for assessing predictions corresponding to measurements with measurement variation. On synthetically generated data that mimics genomics data, fitting with the Deviation Error as the loss function yielded models that were as good if not better than reference models among different combinations of values, metrics, and levels of noise. It was the only loss function of those tried that yielded models that were not significantly worse than those by a reference loss function for any the combinations. Thus, it is also recommended that the Deviation Error be used as the loss function for at least genomics data with measurement variation.

6 Acknowledgements

We thank Anthony J. Bonner and Alan M. Moses from the University of Toronto for fruitful conversations and suggestions.

7 Author Contributions

Conceptualization: Husam Abdulnabi

Data curation: Husam Abdulnabi

Formal analysis: Husam Abdulnabi

Investigation: Husam Abdulnabi

Methodology: Husam Abdulnabi

Project administration: Husam Abdulnabi

Software: Husam Abdulnabi

Supervision: J. Timothy Westwood

Writing – original draft: Husam Abdulnabi

Writing – review & editing: J. Timothy Westwood

8 References

Alipanahi, B., Delong, A., Weirauch, M. T., & Frey, B. J. (2015). Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, *33*(8), 831–838. https://doi.org/10.1038/nbt.3300

Amemiya, H. M., Kundaje, A., & Boyle, A. P. (2019). The ENCODE Blacklist: Identification of Problematic Regions of the Genome. *Scientific Reports*, *9*(1), 9354. https://doi.org/10.1038/s41598-019-45839-z

Avsec, Ž., Weilert, M., Shrikumar, A., Krueger, S., Alexandari, A., Dalal, K., Fropf, R., McAnany, C., Gagneur, J., Kundaje, A., & Zeitlinger, J. (2021). Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, *53*(3), 354–366. https://doi.org/10.1038/s41588-021-00782-6

Barati, B., Erfaninejad, M., & Khanbabaei, H. (2025). Evaluation of effect of optimizers and loss functions on prediction accuracy of brain tumor type using a Light neural network. *Biomedical Signal Processing and Control*, *103*, 107409. https://doi.org/10.1016/j.bspc.2024.107409

Basu, A. (1998). Robust and efficient estimation by minimising a density power divergence. *Biometrika*, *85*(3), 549–559. https://doi.org/10.1093/biomet/85.3.549

Beaton, A. E., & Tukey, J. W. (1974). The Fitting of Power Series, Meaning Polynomials, Illustrated on Band-Spectroscopic Data. *Technometrics*, *16*(2), 147–185. https://doi.org/10.1080/00401706.1974.10489171

Casella, G., & Berger, R. (2002). *Statistical Inference* (2nd ed.). Thomson Learning. https://doi.org/10.1201/9781003456285

Efron, B., & Tibshirani, R. (1986). Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Measures of Statistical Accuracy. *Statistical Science*, *1*(1), 54–57.

Huber, P. J. (1964). Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, *35*(1), 73–101. https://doi.org/10.1214/aoms/1177703732

Kelley, D. R. (2020). Cross-species regulatory sequence activity prediction. *PLoS Computational Biology*, *16*(7), e1008050. https://doi.org/10.1371/journal.pcbi.1008050

Kendall, A., & Gal, Y. (2017). *What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?*

Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*.

Li, H., Xu, Z., Taylor, G., Studer, C., & Goldstein, T. (2018). *Visualizing the Loss Landscape of Neural Nets*.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal Loss for Dense Object Detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2999–3007. https://doi.org/10.1109/ICCV.2017.324

Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, *15*(12), 550. https://doi.org/10.1186/s13059-014-0550-8

McCarthy, D. J., Chen, Y., & Smyth, G. K. (2012). Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Research*, *40*(10), 4288–4297. https://doi.org/10.1093/nar/gks042

Murphy, K. P. (2022). *Probabilistic machine learning: an introduction*. MIT Press.

Nix, D. A., & Weigend, A. S. (1994). Estimating the mean and variance of the target probability distribution. *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, 55–60 vol.1. https://doi.org/10.1109/ICNN.1994.374138

Steininger, M., Kobs, K., Davidson, P., Krause, A., & Hotho, A. (2021). Density-based weighting for imbalanced regression. *Machine Learning*, *110*(8), 2187–2211. https://doi.org/10.1007/s10994-021-06023-5

Tsallis, C. (1988). Possible generalization of Boltzmann-Gibbs statistics. *Journal of Statistical Physics*, *52*(1–2), 479–487. https://doi.org/10.1007/BF01016429

9 Supplementary

9.1 Equations

Sup. Eq. 1. Relative Probability of a Probability Density Function

$$\frac{\Pr(y_1 - \varepsilon \leq y_1 \leq y_1 + \varepsilon)}{\Pr(y_2 - \varepsilon \leq y_2 \leq y_2 + \varepsilon)} = \frac{P(y_1)}{P(y_2)}, P(y_2) > 0$$

where:

$\varepsilon =$ a constant

Sup. Eq. 2. Gamma Distribution

$$G(x; \alpha, \theta) = \frac{(x^{\alpha-1})\left(e^{-\frac{x}{\theta}}\right)}{\Gamma(\alpha)(\theta^{\alpha})}, \alpha > 0, \theta > 0$$

where:

$\Gamma(x) =$ gamma function

$\alpha =$ “shape” parameter

$\theta =$ “scale” parameter

Sup. Eq. 3. Beta Prime distribution

$$H(x; \alpha, \beta) = \frac{(x^{\alpha-1})\big((1+x)^{-\alpha-\beta}\big)}{\mathrm{B}(\alpha, \beta)}, x > 0, \alpha > 0, \beta > 0$$

where:

$$\mathrm{B}(\alpha, \beta) = \frac{\big(\Gamma(\alpha)\big)\big(\Gamma(\beta)\big)}{\Gamma(\alpha + \beta)}$$

Sup. Eq. 4. Ratio of two Gamma variables is a Beta Prime variable

$$\frac{G(x;\alpha_1,\theta_1)}{G(x;\alpha_2,\theta_2)} = \frac{\theta_1}{\theta_2} H(x,\alpha_1,\alpha_2), \theta_2 > 0$$

Sup. Eq. 5. Mode of Beta Prime distribution

$$\text{mode}\big(H(x,\alpha,\beta)\big) = \begin{cases} \dfrac{\alpha-1}{\beta+1}, & \alpha \geq 1 \\ 0, & \text{otherwise} \end{cases} \quad \alpha > 0, \beta > 0$$

Sup. Eq. 6. Homoscedastic Gaussian Noise

$$y_{noised} = y_{original} + N(0,\sigma)$$

where:

$y_{original} =$ original measurement

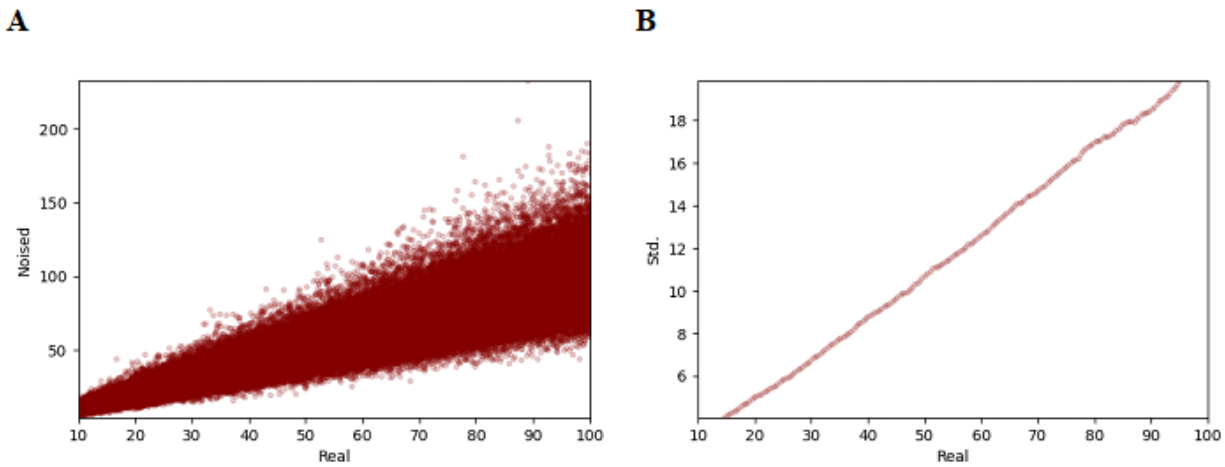
$y_{noised} =$ measurement with noise

$N(\mu,\sigma) =$ Normal distribution with μ mean and standard deviation (σ)

Sup. Eq. 7. Log-scale Homoscedastic Gaussian Noise

$$y_{noised} = y_{original} * \left(e^{N(0,\sigma)}\right)$$

9.2 Figures



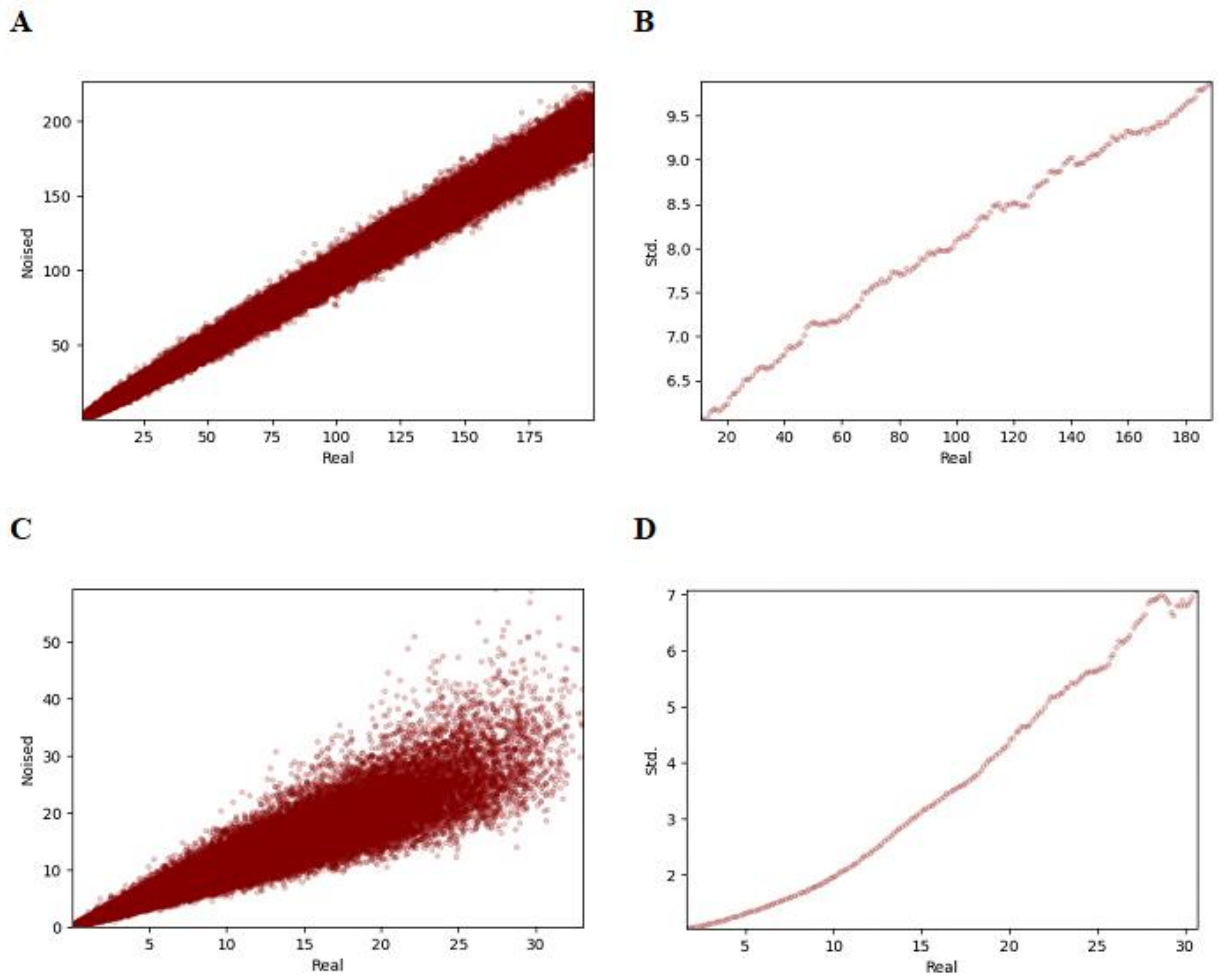
Sup. Fig. 1. Log-scale Gaussian noise.

Real values (100,000 points) were drawn from a uniform distribution with a minimum 10 and a maximum 100. Noise for each actual value was drawn from a normal distribution with a mean 0 and a standard deviation 0.2.

A. Noise was applied to produce log-scale Gaussian noise (**Sup. Eq. 7**) producing "Noised" values.

B. The mean and standard deviations of sliding windows with a size 0.1x and stride of 0.005x the total range of the Noised values in **A**.

The windowed standard deviations of the log-scale Gaussian noise increase linearly.



Sup. Fig. 2. Measurement variation explained by GDs and BPDs resemble log-scale Gaussian noise.

Mimicking genomics measurements, 100,000 target-control pairs were generated. The *real* target values were drawn from a uniform distribution with a minimum of 1 and a maximum of 200. The real control values were drawn from a uniform distribution with a minimum of 5 and a maximum of 20. The modes of the real target-control pairs for BPDs were determined. For each of the real target and real control values, 3 samples were drawn from a GD with an alpha of the real value and a scale of 1 and averaged. The modes of sampled target-control pairs for BPDs were determined.

A, C) Scatter plots compare the real and the noised values for the target (**A**) which follow GDs and for the target-control pairs which follow BPDs (**C**).

B, D) Scatter plots comparing the mean and standard deviations of sliding windows with a size 0.1x and stride of 0.005x the total range of the Noised values in **A** (**B**) and in **C** (**D**).

The windowed standard deviations of the measurements with measurement variation that is explained by GDs and BPDs increase as the real value increases.

9.3 Notes

9.3.1 Model Fitting

NNs are fit using data in the training set and in cycles called epochs. The number of epochs used in fitting can be set prior to modelling. However, more commonly, fitting continues until the model's performance on data in a set other than the training and testing sets – commonly known as the validation set but is referred to here as the stoppage set – deteriorates. An extra number of epochs is allotted to allow the model to regain performance, known as patience. The performance on the stoppage set is assessed by a function known as the stop criteria which is the loss function in this study.

9.4 Tables

All tables in **DevLoss.Sup.Tables.xlsx.**

Sup. Table 1. Mean pairwise bootstrap scores comparing fitting with Adam learning rates from 0.0001 to 0.05 to models fit with an Adam learning rate of 0.001.

Models were fit with MSE as the loss function.

Negative values indicate that models fit using the subject configuration performed better than models fit using the reference configuration. Bolded values indicate that the models fit using the subject configuration are significantly better or worse than models fit using the reference configuration, respectively.

Sup. Table 2. Statistics of pairwise bootstrap scores comparing fitting with different loss functions to fitting with a reference loss function which is the same as the evaluating metric (e.g. the reference loss function is the NLL when evaluating with the NLL). The predictions of the models were compared to the true control-scaled read count with MSE and MAE and with the post-seq control-scaled read count with NLL and MDE.

The "Shuffled MSE" is models fit with MSE as the loss functions but with the outputs in the training and validation sets shuffled, serving as a negative control.

"S.E.", "Conf. Int.", and "Sig." refer to standard error, confidence interval, and significance, respectively. Bolded rows indicate rows that are significant.