

Supplementary Methods: MPRAAnalyze model description

Introduction

MPRAAnalyze aims to infer the true induced transcription rate α for each of the candidate enhancers included in a given MPRA dataset, by modeling the raw counts and avoiding ratio-based summary statistics.

Formally, the MPRA data for a given enhancer can be denoted as $(\vec{d}, \vec{r})$ where $\vec{d} = \begin{bmatrix} d_1 \\ \vdots \\ d_m \end{bmatrix}$ and $\vec{r} = \begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix}$, Where n is the number of available RNA observations and m is the number of available

DNA observations. Note that ideally, the RNA and DNA libraries would both be from the same post-transduction populations, which would allow complete modeling of technical and confounding effects in all layers of the model. However, since many MPRA libraries sequences the DNA from the pre-transduction plasmid population, MPRAAnalyze supports both options.

Regression Model

MPRAAnalyze assumes a linear relationship between RNA and DNA, an assumption easily checked and mostly supported by MPRA data. Formally, we assume that for some transcription rate α , the true DNA and RNA measures follow a linear relationship with slope α :

$$RNA = \alpha \cdot DNA$$

However, the data is usually generated by a set of different distributions which depend on wanted sources of variation like different cell types or cellular environments, or unwanted sources of variation like batch effects or barcode-specific effects. To enable controlling these factors in the analysis, MPRAAnalyze assumes all effects are independent and additive in log space and implements a flexible and extendable regression model.

We construct a separate regression model for the DNA and RNA models, and fit them by likelihood maximization.

Formally, we model each observation as follows:

$$\begin{aligned}\hat{d}_i &= \left(\prod_{\ell} \exp(\beta_{\ell(i)}) \right) \cdot S_{d,i} \Rightarrow \log(\hat{d}_i) = \sum_{\ell} \beta_{\ell(i)} + \log(S_{d,i}) \\ \hat{r}_i &= \hat{\alpha} \hat{d}_i \cdot S_{r,i} = \left(\prod_m \exp(\gamma_{m(i)}) \right) \cdot \left(\prod_{\ell} \exp(\beta_{\ell(i)}) \right) \cdot S_{r,i} \Rightarrow \\ &\Rightarrow \log(\hat{r}_i) = \sum_{\ell} \beta_{\ell(i)} + \sum_m \gamma_{m(i)} + \log(S_{r,i})\end{aligned}$$

Where $S_{d,i}, S_{r,i}$ are library depth normalization factors, ℓ, m relate the observation index to the appropriate batch factor, and β, γ are the set of parameters to optimize.

In this framework, $\prod_{\ell} \beta_{\ell(i)}$ is the estimate of the true number of DNA molecules, and $\prod_m \gamma_{m(i)}$ is the estimate of α , the transcription rate. Note that library depth factors are purely technical and therefore $S_{d,i}$ is only corrected for when fitting the DNA model and not included in the RNA model, and that these factors can be estimated externally. Fitting the parameters across the observations can be done by matrix multiplication with binary design matrices X_d, X_r , simplifying to:

$$\begin{aligned}\log(\vec{\hat{d}}) &= X_d \vec{\beta} + \log(\vec{S}_d) \\ \log(\vec{\hat{r}}) &= X_d \vec{\beta} + X_r \vec{\gamma} + \log(\vec{S}_r)\end{aligned}$$

We then optimize these parameters by optimizing the likelihood function of the data, conditioned on some distributional assumptions (detailed below). Formally we use the vector of estimates $\vec{\hat{d}}, \vec{\hat{r}}$ as estimators for the first moment of the distribution the observations were generated by. The exact likelihood computation depends on the distributional model, but conceptually we get:

$$\begin{aligned}(\beta^{MLE}, \gamma^{MLE}) &= \arg \max_{\vec{\beta}, \vec{\gamma}} \left\{ \mathcal{L}_D(\vec{d} | \vec{\hat{d}}) \cdot \mathcal{L}_R(\vec{r} | \vec{\hat{r}}) \right\} \\ &= \arg \max_{\vec{\beta}, \vec{\gamma}} \left\{ \mathcal{L}_D(\vec{d} | \vec{\hat{d}}) \cdot \mathcal{L}_R(\vec{r} | \vec{\hat{d}} \cdot \hat{\alpha} \cdot \vec{S}_r) \right\} \\ &= \arg \max_{\vec{\beta}, \vec{\gamma}} \left\{ \ell_D(\vec{d} | \exp(X_d \vec{\beta} + \log(\vec{S}_d))) + \ell_R(\vec{r} | \exp(X_d \vec{\beta} + X_r \vec{\gamma} + \log(\vec{S}_r))) \right\} \\ &= \arg \max_{\vec{\beta}, \vec{\gamma}} \left\{ \ell_D(\vec{d} | \exp(X_d \vec{\beta}) \cdot \vec{S}_d) + \ell_R(\vec{r} | \exp(X_d \vec{\beta}) \cdot \exp(X_r \vec{\gamma}) \cdot \vec{S}_r) \right\}\end{aligned}$$

With $\mathcal{L}_D, \mathcal{L}_R$ are the likelihood functions of the DNA, RNA distributions (respectively), and ℓ_D, ℓ_R their respective log-likelihood functions.

Distributional Models

MPRAnalyze models the generation of the data as:

$$\vec{r} = \alpha \vec{d}$$

The main complexity of the model therefore stems from the RNA observations being a convolution of the scaled DNA distribution and the RNA measurement distribution. We offer two ways to handle this:

1 Closed Form Solution

Some distributions provide a closed form for this scenario. MPRAnalyze uses this by assuming a Gamma-Poisson model: we assume the DNA model follows a Gamma distribution: $\vec{d} \sim \text{Gamma}(k, b)$ and that $\vec{r}|\vec{d} \sim \text{Poisson}(\alpha \cdot \vec{d})$. The resulting likelihood for the RNA model has a closed form as a Negative-Binomial distribution, which is a well established approximation of sequencing data. Formally:

$$\begin{aligned} \vec{d} &\sim \text{Gamma}(k, \beta) \\ \Rightarrow \alpha \cdot \vec{d} &\sim \text{Gamma}\left(k, \frac{\beta}{\alpha}\right) \\ \vec{r}|\vec{d} &\sim \text{Poisson}(\alpha \cdot \vec{d}) \\ \Rightarrow \vec{r} &\sim \text{Negative} - \text{Binomial}\left(\mu = \frac{\alpha \cdot k}{\beta}, \psi = k\right) \end{aligned}$$

To incorporate the regression scheme described above in this setting, we share the dispersion parameter a between the DNA and RNA models. The DNA regression model is the estimate for $\left(\frac{1}{b}\right)$, and the RNA regression model is the estimate for α . Therefore, the MLE estimates are:

$$\begin{aligned} (\beta^{MLE}, \gamma^{MLE}, a^{MLE}) &= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\alpha}} \left\{ \mathcal{L}_{\text{Gamma}}(\vec{d} \mid a = \hat{a}, b = \hat{b}) \cdot \mathcal{L}_{NB}(\vec{r} \mid \mu = \frac{\hat{a} \cdot \hat{\alpha}}{\hat{b}}, \psi = \hat{a}) \right\} \\ &= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\alpha}} \left\{ \ell_{\text{Gamma}}\left(\vec{d} \mid a = \hat{a}, b = \left(\exp(X_d \vec{\beta}) \cdot \vec{S}_D\right)^{-1}\right) \right. \\ &\quad \left. + \ell_{NB}(\vec{r} \mid \mu = \hat{a} \cdot \exp(X_d \vec{\beta}) \cdot \exp(X_r \vec{\gamma}) \cdot \vec{S}_R, \psi = \hat{a}) \right\} \end{aligned}$$

2 Point estimator

Alternatively, the model can be simplified by fitting the DNA model and collapsing it to a point estimator. Unlike the closed form solution, this means that not all of the information of the DNA model will be shared with the RNA model. However, this means that this option allows for more flexibility in the distributional assumptions, essentially allowing any pair of distributions to be used for the DNA and RNA models. This flexibility is mostly important for the restrictions imposed on the variance estimation. MPRAnalyze implements this approach in two ways, allowing for two sets of constraints on the variance.

2.1 log-Normal DNA, Negative-Binomial RNA

assuming:

$$\begin{aligned} \vec{d} &\sim \log - \text{Normal}(m, \sigma^2) \\ \vec{r} &\sim \text{Negative} - \text{Binomial}(\mu = \alpha m, \psi) \end{aligned}$$

Which, along with the regression scheme, translates to:

$$\begin{aligned}
(\beta^{MLE}, \gamma^{MLE}, \sigma^{MLE}, \psi^{MLE}) &= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\sigma}, \hat{\psi}} \left\{ \mathcal{L}_{logNormal}(\vec{d} \mid m = \hat{m}, \sigma^2 = \hat{\sigma}^2) \cdot \mathcal{L}_{NB}(\vec{r} \mid \mu = \hat{m} \cdot \hat{\alpha}, \psi = \hat{\psi}) \right\} \\
&= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\sigma}, \hat{\psi}} \left\{ \ell_{logNormal}(\vec{d} \mid m = \exp(X_d \vec{\beta}) \cdot \vec{S}_D, \sigma^2 = \hat{\sigma}^2) \right. \\
&\quad \left. + \ell_{NB}(\vec{r} \mid \mu = \exp(X_d \vec{\beta}) \cdot \exp(X_r \vec{\gamma}) \cdot \vec{S}_R, \psi = \hat{\psi}) \right\}
\end{aligned}$$

2.2 log-Normal DNA, log-Normal RNA

assuming:

$$\begin{aligned}
\vec{d} &\sim \log - Normal(m, \sigma_d^2) \\
\vec{r} &\sim \log - Normal(m \cdot \alpha, \sigma_r^2)
\end{aligned}$$

which, along with the regression scheme, translates to:

$$\begin{aligned}
(\beta^{MLE}, \gamma^{MLE}, \sigma_d^{MLE}, \sigma_r^{MLE}) &= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\sigma}_d, \hat{\sigma}_r} \left\{ \mathcal{L}_{logNormal}(\vec{d} \mid \mu = \hat{m}, \sigma^2 = \hat{\sigma}_d^2) \cdot \mathcal{L}_{logNormal}(\vec{r} \mid \mu = \hat{m} \cdot \hat{\alpha}, \sigma^2 = \hat{\sigma}_r^2) \right\} \\
&= \arg \max_{\vec{\beta}, \vec{\gamma}, \hat{\sigma}_d, \hat{\sigma}_r} \left\{ \ell_{logNormal}(\vec{d} \mid \mu = \exp(X_d \vec{\beta}) \cdot \vec{S}_D, \sigma^2 = \hat{\sigma}_d^2) \right. \\
&\quad \left. + \ell_{logNormal}(\vec{r} \mid \mu = \exp(X_d \vec{\beta}) \cdot \exp(X_r \vec{\gamma}) \cdot \vec{S}_R, \sigma^2 = \hat{\sigma}_r^2) \right\}
\end{aligned}$$

Hypothesis testings

MPRAnalyze implements a variety of hypothesis testing options to accommodate the variety of different MPRA experimental design settings. The two major types of analyses MPRA facilitates are quantitative analysis: determining the level of transcriptional activity induced by each candidate enhancer; and comparative analysis: identifying differential activity of a candidate enhancer between biological conditions. MPRAnalyze provides a robust statistical framework that enables addressing these questions while leveraging the existence of negative controls (scramble sequences) in the experiment. The various use cases are summarized by the following diagram, and fully explained below:

Comparative Analysis

A comparative analysis is one where the leading question is about the difference in activity of a given enhancer between conditions. Examples of these include comparing enhancer activity between cell types to detect tissue-specific activity, comparing different time-points in a differentiation process and comparing activity in unstimulated cells with cells exposed to various stimuli. MPRAnalyze offers several possible settings and testing frameworks to address these kinds of questions.

Coefficient-based Testing

A coefficient-based testing framework addresses these comparisons by directly testing the regression coefficients for significance, using a Wald test. This approach enables testing multiple different hypotheses

using the same model fit, which can be useful in some cases. For example, if effects of different stimuli are to be tested, a single model can be fit to the entire dataset, then testing the effect of each stimulus is simply computing the Wald-test for the appropriate regression coefficient, with the unstimulated samples as the reference.

Likelihood-Ratio Testing

A likelihood Ratio test lacks the post-fitting flexibility of a coefficient-based testing, but allows more complex questions to be asked. For instance, looking at a time-series dataset of a differentiation process, one might ask “does this enhancer change their activity over time”. This question can be difficult to address with coefficient-based testing, since each single coefficient only corresponds to one time point. However, this can easily be addressed with a likelihood-ratio test. This is done by fitting two models: a ‘full’ model with the complete design including the conditional factor (in this case, the factors that identify the time-point each data-point is associated with), and another model - a ‘reduced’ model - that is identical to the full model in everything except that it lacks this factor of interest. Since the reduced space is a sub-space of the full space (in which the time-related coefficients are 0), these models are nested and a standard Likelihood-Ratio Test can be used to test for statistical significance.

Classification Analysis

A classification analysis address the question of the overall degree of activity of a given enhancer. Note that addressing this question without an established baseline for the basal transcription rate is difficult, and the power and interpretability of these analyses are very limited in the absence of negative controls. MPRAalyze performs classification by testing each enhancer’s estimated transcription rate, α , against the null distribution. This null distribution is based on the *alphas* of the negative controls if available, and on the bottom half of the total distribution otherwise, under the implicit assumption that active enhancers will be more active than the median enhancer. In either case, MPRAalyze computes both a Z-score and a MAD-score, which is a median-based variant of the Z-score. These scores are normally distributed under the null, and we can therefore compute p-values from the standard normal distribution.

Classification with Controls

Negative controls can be used directly to establish the null distribution, then computing the scores within the null. Formally, for $\vec{\alpha}_c = (\alpha_{c_1}, \dots, \alpha_{c_\ell})$ the estimated rates for the ℓ controls included in the experiment:

$$\begin{aligned} \text{MAD-score}_c(\alpha_i) &= \frac{\alpha_i - \text{median}(\vec{\alpha}_c)}{K \cdot \text{median}(|\alpha_{c_i} - \text{median}(\vec{\alpha}_c)|)} \\ \text{Z-score}_c(\alpha_i) &= \frac{\alpha_i - \text{mean}(\vec{\alpha}_c)}{\text{std}(\vec{\alpha}_c)} \end{aligned}$$

Additionally, MPRAalyze also calculates the empirical p-value:

$$P(\alpha_i) = \frac{1}{\ell} \sum_{j=1}^{\ell} 1(\alpha_i > \alpha_{c_j})$$

Classification without Controls

Without the advantage of negative controls to establish the null transcription rate, MPRAnalyze empirically determines the distributional properties of the null set. For this, MPRAnalyze relies on three assumptions: (1) the data contains both active and inactive enhancers; (2) active enhancers generally have a higher estimated rate of transcription than inactive enhancers; (3) not all active enhancers are equally active, but all inactive enhancers are equally inactive, meaning generated by the same distribution determined by the properties of the promoter used. MPRAnalyze empirically estimates the mode of the distribution, by taking the center of the window which contains the maximal number of samples, with a window size set to be 0.02 of the total range of values in the sample. This mode is assumed to be the center of the null distribution, due to the third assumption. Then samples that are lower from the mode are then defined as the null set, and are used to estimate the width of the distribution, assuming the null is symmetric with respect to the mode.

The calculation of the scores is done using the null set:

$$\begin{aligned}\text{MAD-score}_{\text{null}} &= \frac{\alpha_i - \text{mode}(\vec{\alpha})}{K \cdot \text{median}(\text{mode}(\vec{\alpha}) - \alpha_{\text{null}}^{\vec{}})} \\ \text{Z-score}_{\text{null}} &= \frac{\alpha_i - \text{mode}(\vec{\alpha})}{\sqrt{\text{mean}\left((\text{mode}(\vec{\alpha}) - \alpha_{\text{null}}^{\vec{}})^2\right)}}\end{aligned}$$

Incorporating Negative Controls

MPRAnalyze enables incorporating negative controls into the model to correct systemic bias in the data. Such bias often stems from inherent biological differences between condition that are not related to the regulatory mechanism studied, e.g cellular size differences. Including negative controls in the experiment, preferably by including randomized “scrambled” sequences in the experiment, allows quantifying and correcting these effects. This can be done in several ways, all of which involve some manner of fitting a joint model for control enhancers.

All negative controls are assumed to be generated by the same, baseline, RNA model. However, since there’s no reason to assume this has any effect on transfection efficiency, a separate DNA model is still fit to each control enhancer. Formally, for n control enhancers, the joint model is:

$$\begin{aligned} & (\beta_1^{MLE}, \dots, \beta_n^{MLE}, \gamma^{MLE}) = \\ & \arg \max_{\beta_1, \dots, \beta_n, \gamma} \left\{ \sum_{i=1}^n \ell_D \left(\vec{d} \mid \exp \left(X_d \vec{\beta}_i \right) \cdot \vec{S}_d, \psi_D \right) + \sum_{i=1}^n \ell_R \left(\vec{r} \mid \exp \left(X_d \vec{\beta}_i \right) \cdot \exp \left(X_r \vec{\gamma} \right) \cdot \vec{S}_r, \psi_R \right) \right\} \end{aligned}$$

Fitting a joint models of or with controls can be used in several ways to account for systemic bias in the data.

Control-based Correction Factors

The joint control model described above can be used to generate normalization factors based on systemic bias reflected in the

A separate model can be fit for the controls, and correction factors extracted from it and included in the general model. Similar to library-depth correction, this would amount to an additional scaling

factor added to the regression scheme that corrects the inherent bias in the system as measured by the negative controls.

Theoretical Supplements

Gamma-Poisson convolution proof

Proof of the closed form relationship of the likelihood of the RNA model in the Gamma-Poisson model:

Let $D \sim \text{Gamma}(k, \beta)$ and $R|D \sim \text{Poisson}(\lambda = \alpha D)$.

First, we'll note that $\alpha D \sim \text{Gamma}\left(k, \frac{\beta}{\alpha}\right)$, and therefore:

$$\begin{aligned}
P(R = x) &= \int_0^{\infty} P(R = x|\lambda) P_{\alpha D}(\lambda) d\lambda \\
&= \int_0^{\infty} \frac{\lambda^x e^{-\lambda}}{x!} \cdot \frac{\left(\frac{\beta}{\alpha}\right)^k}{\Gamma(k)} \cdot \lambda^{k-1} e^{-\frac{\beta\lambda}{\alpha}} d\lambda \\
&= \frac{\left(\frac{\beta}{\alpha}\right)^k}{\Gamma(k) \cdot x!} \int_0^{\infty} \lambda^{k+x-1} e^{-\lambda\left(\frac{\beta}{\alpha}+1\right)} d\lambda \\
&= \frac{\left(\frac{\beta}{\alpha}\right)^k}{\Gamma(k) \cdot x!} \cdot \left(1 + \frac{\beta}{\alpha}\right)^{-(k+x)} \Gamma(k+x) \\
&= \frac{\Gamma(k+x)}{\Gamma(k) \cdot x!} \cdot \left(\frac{\frac{\beta}{\alpha}}{1 + \frac{\beta}{\alpha}}\right)^k \cdot \left(\frac{1}{1 + \frac{\beta}{\alpha}}\right)^x \\
&\Rightarrow R \sim NB\left(\mu = \frac{\alpha \cdot k}{\beta}, \psi = k\right)
\end{aligned}$$