Transcriptional regulation via TF-modifying enzymes - an integrative model-based analysis
Supporting Information

Logan J. Everett, Shane T. Jensen, Sridhar Hannenhalli

Contents

1 Model and Estimation Method

In this section we introduce and explain the underlying mathematics of our model, and introduce a novel heuristic approach to efficiently estimate the model parameters. The primary motivation of fitting our model to input data is to predict the network of connectivity between modifiers, TFs, and target genes. We use the same notation here as in the main text, with vectors and matrices denoted in bold. See Tables S1 and S2 for a guide to our notation.

1.1 Primary Model Equations

The model is described concisely by the following equations and notation. Given target genes of interest indexed by i from 1 to N, TFs of interest indexed by j from 1 to J, and modifiers of interest indexed by k from 1 to K, with expression measured under conditions indexed by t from 1 to T: let $\mathbf{g}$ denote the $N \times T$ expression matrix for target genes with values g_{it}; let $\mathbf{f}$ denote the $J \times T$ expression matrix for TFs with values f_{jt}; and let $\mathbf{h}$ denote the $K \times T$ expression matrix for modifiers with values h_{kt}. Thus, the first step of the modeling procedure is the selection of an appropriate set of candidate TFs, modifiers, and target genes for the network of interest. In particular, the target genes i should not overlap with the TFs j or modifiers k. The model defines each gene expression value g_{it} as a function of TF and modifier activities:

$$g_{it} = \alpha_i + \sum_{j=1}^{J} \beta_j C_{ij} f_{jt} + \sum_{j=1}^{J} \sum_{k=1}^{K} \gamma_{jk} C_{ij} D_{jk} \Phi(f_{jt}, h_{kt}) + \varepsilon_{it} \tag{1.1}$$

Eq 1.1 is applied to all genes i from 1 to N and all samples t from 1 to T. The term α_i represents a baseline expression value for gene i independent of condition. The term β_j is a scaling factor for TF j to describe its influence on all target genes. The terms C_{ij}, denoted collectively by the $N \times J$ matrix $\mathbf{C}$, are binary variables indicating whether each TF j regulates the expression of each gene i. Modifiers do not influence target genes directly in our model. Rather, they influence target genes indirectly through their effects on TF activity. The terms D_{jk}, denoted collectively by the $J \times K$ matrix $\mathbf{D}$, are binary variables indicating whether each modifier k has a synergistic effect on the activity of each TF j. The nature of these synergistic effects are described in general as a function of expression values, denoted by Φ, discussed in more detail below. The term ε_{it} refers to the residual error of the model fit for each g_{it}. This term captures technical and biological noise as well as non-model behavior and is sampled from the normal distribution with model-wide variance σ^2:

$$\varepsilon_{it} \sim \mathrm{N}(0, \sigma^2) \;\; \forall i, t \tag{1.2}$$

The interaction function Φ is generally intended to be non-linear in order to capture "synergistic" effects of TF-Modifier relationships. The function can be applied prior to model fitting and therefore does not alter the complexity of the computations discussed in subsequent sections. We use the following function in order to capture a synergistic effect

which is positive only when both input expression values are positive, and negative in all other cases:

$$\Phi(f,h) = \text{sign}(\min(f,h))|f*h| = \begin{cases} f*h & \text{iff } f,h>0 \\ -|f*h| & \text{otherwise} \end{cases} \tag{1.3}$$

This function is intended for normalized expression profiles with mean 0 (see Sec 2), and has the preferable behavior that it returns positive values only when f and h are both positive (high expression). This function is based on the biological intuition that synergistic effects only occur when both members are sufficiently expressed, and has been successful in our applications. However, it does not have a symmetric intuition for negative synergistic effects, i.e. when $\gamma_{jk} < 0$, this intuition may be lost. Ideally, we want a non-linear function that is also monotonic in both f and h, and which fits our biological intuitions for both positive and negative values of γ_{jk}. We leave the identification of such a function for future work, but also note that the rest of our methodology is independent of the choice made for this function.

The model also allows the consideration of other types of high-throughput data, in addition to expression. Let $\mathbf{b}$ and $\mathbf{m}$ denote two $N \times J$ matrices of prior probabilities for $\mathbf{C}$ based on different types of biological data, i.e. using binding data (i.e. ChIP-chip or ChIP-seq) for $\mathbf{b}$ and Positional Weight Matrices (PWMs) applied to promoter sequence data for $\mathbf{m}$. The entries in these matrices are denoted b_{ij} and m_{ij} respectively. Likewise, $J \times K$ matrices $\mathbf{a}$ (protein array, functional association, or interaction data) and $\mathbf{s}$ (substrate-prediction) can be defined as priors for $\mathbf{D}$. The variable matrices $\mathbf{C}$ and $\mathbf{D}$ in the model are given prior probabilities based on a weighted mixture of all available biological priors, as follows:

$$P(C_{ij}=1) = b_{ij}^{w_j} m_{ij}^{(1-w_j)} \tag{1.4}$$

$$P(D_{jk}=1) = a_{jk}^{u_k} s_{jk}^{(1-u_k)} \tag{1.5}$$

The weight variables, denoted collectively as $\mathbf{w}$ and $\mathbf{u}$, each range from 0 to 1 and are also estimated as part of the model-fitting procedure. Thus, no prior assumptions need to be made about the relative quality of each prior source. Note that there are separate weight variables w_j and u_k for each TF j and modifier k respectively. This was chosen because the quality of a given PWM or ChIP-chip result set will typically vary more widely by TF, rather than by target gene. Likewise, the PWM or interaction assay data for modifiers is also assumed to vary more widely by modifier, rather than by substrate or interaction partner. When only a single type of prior is available for either $\mathbf{C}$ or $\mathbf{D}$, the equations above simplify to an unweighted use of the priors.

Also note that given any full instantiation of values for $\mathbf{C}$ and $\mathbf{D}$, Eq 1.1 becomes a linear regression and can be solved using a standard closed-form solution. Although this problem can be solved as a linear regression, the relationship between target gene, TF, and modifier expression values is non-linear because of the synergy function Φ.

1.2 Model Posterior

As in the main text, let Θ denote the set of model parameters $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \sigma, \mathbf{w}, \mathbf{u})$, excluding $\mathbf{C}$ and $\mathbf{D}$. Let Ξ denote the complete set of biological data sources $(\mathbf{g}, \mathbf{f}, \mathbf{h}, \mathbf{b}, \mathbf{m}, \mathbf{a}, \mathbf{s})$. Thus, the model can be completely described by the set $(\Xi, \Theta, \mathbf{C}, \mathbf{D})$. The relative posterior probability of a model configuration, given a set of biological data is:

$$P(\Theta, \mathbf{C}, \mathbf{D}|\Xi) \propto P(\mathbf{g}|\mathbf{f}, \mathbf{h}, \mathbf{C}, \mathbf{D}, \Theta) * P(\mathbf{C}|\mathbf{m}, \mathbf{b}, \mathbf{w}) * P(\mathbf{D}|\mathbf{a}, \mathbf{s}, \mathbf{u}) * P(\Theta) \tag{1.6}$$

Or, put another way, the posterior probability of a model given data is proportional to the product of the gene expression likelihood, the edge likelihoods, and the remaining parameter priors. These terms are further defined as:

$$\begin{aligned}
P(\mathbf{g}|\mathbf{f}, \mathbf{h}, \mathbf{C}, \mathbf{D}, \Theta) &= \prod_{i=1}^{N}\prod_{t=1}^{T}(2\pi\sigma^2)^{-1/2}\exp\left[\frac{-1}{2\sigma^2}\varepsilon_{it}^2\right] \\
P(\mathbf{C}|\mathbf{m}, \mathbf{b}, \mathbf{w}) &= \prod_{i=1}^{N}\prod_{j=1}^{J}\left[b_{ij}^{C_{ij}}(1-b_{ij})^{1-C_{ij}}\right]^{w_j}\left[m_{ij}^{C_{ij}}(1-m_{ij})^{1-C_{ij}}\right]^{1-w_j} \\
P(\mathbf{D}|\mathbf{a}, \mathbf{s}, \mathbf{u}) &= \prod_{j=1}^{J}\prod_{k=1}^{K}\left[a_{jk}^{D_{jk}}(1-a_{jk})^{1-D_{jk}}\right]^{u_k}\left[s_{jk}^{D_{jk}}(1-s_{jk})^{1-D_{jk}}\right]^{1-u_k} \\
P(\Theta) &= \prod_{i=1}^{N}(\tau_\alpha^2)^{-1/2}\exp\left[\frac{-1}{2\tau_\alpha^2}\alpha_i^2\right] * \prod_{j=1}^{J}(\tau_\beta^2)^{-1/2}\exp\left[\frac{-1}{2\tau_\beta^2}\beta_j^2\right] \\
&\quad * \prod_{j=1}^{J}\prod_{k=1}^{K}(\tau_\gamma^2)^{-1/2}\exp\left[\frac{-1}{2\tau_\gamma^2}\gamma_{jk}^2\right] * (\sigma^2)^{-2}\exp\left[\frac{-1}{2\sigma^2}\right]
\end{aligned}$$

The priors on all model parameters in Θ are the same as in [1]. $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ are assumed to have normal priors with standard deviations τ_α, τ_β, and τ_γ respectively. The τ hyper-parameters are set to large values, i.e. 10000, to make the parameter priors uninformative. The model-wide variance, σ^2, is assumed to have a prior defined by the χ_ν^2 distribution with $\nu = 2$. The weight variables $\mathbf{w}$ and $\mathbf{u}$ are given uniform priors in the range $(0, 1)$.

1.3 Individual Parameter Posteriors

Solving for the full posterior distribution (Eq 1.6) analytically is an intractable problem. However, it is possible to compute an "individual posterior" on each model parameter given the biological data *and* fixed values for all other model parameters. These equations form the basis for iterative approaches to estimating the model parameters.

First, let $\Theta_{\{-Q\}}$ denote the set of all parameters in Θ *except* some individual parameter Q. Let $\varepsilon'_{it}[Q = q]$ denote the residual error of gene i in sample t when some model parameter Q is changed to value q. The posterior distribution for a particular α_i given all expression

data and all other model parameters is:

$$P\left(\alpha_i = q|\Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\alpha_i\}}\right) \propto \exp\left[\frac{-1}{2\sigma^2}\sum_{t=1}^{T}\varepsilon'_{it}[\alpha_i = q]^2\right] * \exp\left[\frac{-1}{2\tau_\alpha^2}q^2\right] \tag{1.7}$$

This distribution can be sampled as a normal distribution with mean μ_{α_i} and variance ν_α:

$$\mu_{\alpha_i} = \frac{\nu_\alpha}{\sigma^2} * \sum_{t=1}^{T}\varepsilon'_{it}[\alpha_i = 0] \tag{1.8}$$

$$\nu_\alpha = (T/\sigma^2 + 1/\tau_\alpha^2)^{-1} \tag{1.9}$$

Alternatively, the value of α_i can be chosen to maximize the posterior by setting $\alpha_i = \mu_{\alpha_i}$. Also note that for an "unregulated" gene, i.e. a gene i such that $C_{ij} = 0 \ \ \forall j$, Eq 1.8 simplifies to an estimate of the mean expression for gene i:

$$\mu_\alpha = \frac{1}{T + \sigma^2/\tau_\alpha^2}\sum_{t=1}^{T} g_{it}$$

Similarly, the individual posterior for β_j is:

$$P(\beta_j = q|\Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\beta_j\}}) \propto \exp\left[\frac{-1}{2\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[\beta_j = q]^2\right] * \exp\left[\frac{-1}{2\tau_\beta^2}q^2\right] \tag{1.10}$$

Therefore, we can resample $\beta_j \sim \mathrm{N}(\mu_{\beta_j}, \nu_{\beta_j})$ with mean and variance defined as:

$$\mu_{\beta_j} = \frac{\nu_{\beta_j}}{\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[\beta_j = 0] * C_{ij}f_{jt} \tag{1.11}$$

$$\nu_{\beta_j} = \left(\frac{\sum_{i=1}^{N}\sum_{t=1}^{T}(C_{ij}f_{jt})^2}{\sigma^2} + \frac{1}{\tau_\beta^2}\right)^{-1} \tag{1.12}$$

Once again, the posterior can be maximized by setting $\beta_j = \mu_{\beta_j}$. Also note that $\mu_{\beta_j} = 0$ and $\nu_{\beta_j} = \tau_\beta^2$ when $C_{ij} = 0 \ \ \forall i$. In other words, when TF j does not regulate any genes, its posterior becomes equivalent to its uninformative prior. Similar equations are derived below for resampling or maximizing the individual posterior of γ_{jk}. In this case, the posterior is equivalent to the the uninformative prior whenever $C_{ij} = 0 \ \ \forall i$ or when $D_{jk} = 0$.

$$P(\gamma_{jk} = q|\Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\gamma_{jk}\}}) \propto \exp\left[\frac{-1}{2\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[\gamma_{jk} = q]^2\right] * \exp\left[\frac{-1}{2\tau_\gamma^2}q^2\right] \tag{1.13}$$

$$\mu_{\gamma_{jk}} = \frac{\nu_{\gamma_{jk}}}{\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[\gamma_{jk} = 0] * C_{ij}D_{jk}\Phi(f_{jt}, h_{kt}) \tag{1.14}$$

$$\nu_{\gamma_{jk}} = \left(\frac{\sum_{i=1}^{N}\sum_{t=1}^{T}\left(C_{ij}D_{jk}\Phi(f_{jt}, h_{kt})\right)^2}{\sigma^2} + \frac{1}{\tau_\gamma^2}\right)^{-1} \tag{1.15}$$

The individual posterior for the model-wide residual error variance, σ^2, given all input data and all other parameters is:

$$P(\sigma^2|\Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\sigma^2\}}) \propto (\sigma^2)^{-\left(\frac{TN}{2}+2\right)} * \exp\left[\frac{-1}{2\sigma^2}\sum_{i=1}^{N}\sum t = 1^T \varepsilon_{it}^2\right] * \exp\left[\frac{-1}{2\sigma^2}\right] \quad (1.16)$$

The above equation implies that σ^2 can be sampled from a scaled-inverse χ^2 distribution with degrees of freedom parameter $TN+2$ and scale parameter s^2 defined in Eq 1.17 below. This posterior can be maximized by setting $\sigma^2 = s^2$.

$$s^2 = \frac{1 + \sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon_{it}^2}{TN+2} \quad (1.17)$$

The $\mathbf{C}$ and $\mathbf{D}$ variables are binary, and therefore we only need to determine the relative probability of values 1 and 0 for each variable. For C_{ij}, the individual posterior is given by Eq 1.18 for $q \in \{0, 1\}$:

$$\begin{aligned} P(C_{ij} = q|\Xi, \mathbf{C}_{\{-C_{ij}\}}, \mathbf{D}, \Theta) \quad \propto \quad & \exp\left[\frac{-1}{-2\sigma^2}\sum_{t=1}^{T}\varepsilon'_{it}[C_{ij} = q]^2\right] \\ & * \left[b_{ij}^q(1-b_{ij})^{1-q}\right]^{w_j} * \left[m_{ij}^q(1-m_{ij})^{1-q}\right]^{1-w_j} \end{aligned} \quad (1.18)$$

To find the exact posteriors for $C_{ij} = 0$ and $C_{ij} = 1$, we first compute the proportional probability values $Z_0^{(C_{ij})}$ and $Z_1^{(C_{ij})}$, respectively:

$$Z_0^{(C_{ij})} = \exp\left[\frac{-1}{-2\sigma^2}\sum_{t=1}^{T}\varepsilon'_{it}[C_{ij} = 0]^2\right] * (1-b_{ij})^{w_j}(1-m_{ij})^{1-w_j} \quad (1.19)$$

$$Z_1^{(C_{ij})} = \exp\left[\frac{-1}{-2\sigma^2}\sum_{t=1}^{T}\varepsilon'_{it}[C_{ij} = 1]^2\right] * b_{ij}^{w_j} m_{ij}^{1-w_j} \quad (1.20)$$

Therefore, C_{ij} can be resampled by setting $C_{ij} = 1$ with probability $\frac{Z_1^{(C_{ij})}}{Z_0^{(C_{ij})}+Z_1^{(C_{ij})}}$ and can be maximized by setting $C_{ij} = 1$ iff $Z_1^{(C_{ij})} > Z_0^{(C_{ij})}$. Using a similar posterior equation for D_{jk} yields $\zeta_0^{(D_{jk})}$ and $\zeta_1^{(D_{jk})}$ analogous to $Z_0^{(C_{ij})}$ and $Z_1^{(C_{ij})}$ respectively:

$$\zeta_0^{(D_{jk})} = \exp\left[\frac{-1}{-2\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[D_{jk} = 0]^2\right] * (1-a_{jk})^{u_k}(1-s_{jk})^{1-u_k} \quad (1.21)$$

$$\zeta_1^{(D_{jk})} = \exp\left[\frac{-1}{-2\sigma^2}\sum_{i=1}^{N}\sum_{t=1}^{T}\varepsilon'_{it}[D_{jk} = 1]^2\right] * a_{jk}^{u_k} s_{jk}^{1-u_k} \quad (1.22)$$

The individual posteriors of the weight parameters w_j and u_k are independent of the expression data and other parameters because all edge variables in $\mathbf{C}$ and $\mathbf{D}$ have defined values. The individual posterior distributions are calculated as follows:

$$P(w_j|\mathbf{C},\mathbf{b},\mathbf{m}) \propto \frac{\left[\prod_{i=1}^{N} b_{ij}^{C_{ij}}(1-b_{ij})^{1-C_{ij}}\right]^{w_j} * \left[\prod_{i=1}^{N} m_{ij}^{C_{ij}}(1-m_{ij})^{1-C_{ij}}\right]^{1-w_j}}{\prod_{i=1}^{N}\left[b_{ij}^{w_j} m_{ij}^{1-w_j} + (1-b_{ij})^{w_j}(1-m_{ij})^{1-w_j}\right]} \tag{1.23}$$

$$P(u_k|\mathbf{D},\mathbf{a},\mathbf{s}) \propto \frac{\left[\prod_{j=1}^{J} a_{jk}^{D_{jk}}(1-a_{jk})^{1-D_{jk}}\right]^{u_k} * \left[\prod_{j=1}^{J} s_{jk}^{D_{jk}}(1-s_{jk})^{1-D_{jk}}\right]^{1-u_k}}{\prod_{j=1}^{J}\left[a_{jk}^{u_k} s_{jk}^{1-u_k} + (1-a_{jk})^{u_k}(1-s_{jk})^{1-u_k}\right]} \tag{1.24}$$

We use grid sampling to estimate the distribution of w_j or u_k. We describe the process, without loss of generality, for a single parameter w_j. First, we define $\hat{p}_\lambda = P(w_j = \lambda|\mathbf{C},\mathbf{b},\mathbf{m})$ (Eq 1.23) for all λ values in $\{0.01, 0.02, \ldots, 0.99\}$. These values are then normalized to probabilities: $p_\lambda = \hat{p}_\lambda / \sum_\lambda \hat{p}_\lambda$. We then sample w_j from all possible values of λ, each with probability p_λ. Alternatively, we can maximize the posterior of w_j by selecting the $w_j = \arg\max_\lambda (p_\lambda)$.

1.4 Model Estimation Method

The equations described above can be used in a standard iterative framework, such as Gibbs Sampling [2] or hill-climbing. However, the posterior distributions relevant to this model are typically highly multi-modal. This results in many regions of the parameter space that are "locally good", i.e. they are considerably more likely than similar model configurations. In other words, there are often multiple solutions for fitting the model to a given set of input data, which are substantially different and roughly equal in their "goodness of fit". This presents a problem for normal statistical learning techniques, as maximization-based techniques are only guaranteed to find local optima, and sampling techniques require an impractical number of iterations to fully explore such a parameter space.

We present here a heuristic approach that combines several statistical learning approaches to produce a good fit of the network model to the input data, with partial or local estimates for key variables of interest. This algorithm can subsequently be run multiple times to more fully explore the solution space in a scalable and robust manner. Note that each run of this algorithm explores only the local solution space around a single mode. The problem of multiple solution modes is handled when combining estimates from multiple runs of this algorithm, and this step is discussed in more depth in Sec 1.5.

The algorithm presented here is run in 3 phases, described below. In the first phase, a fixed starting network is selected and other model parameters are fit to this network using a closed form solution. In the second phase, all model parameters, including network edges, are iteratively maximized until sufficient convergence at a local optimum. In the third phase, model parameters of interest are iteratively resampled a fixed number of times to provide a more robust estimate around the local optimum.

Phase 1: Model Initialization

To seed the algorithm, an initial network $(\mathbf{C}, \mathbf{D})$ is selected. The simplest way to select this network is deterministically, by setting a threshold q on the geometric mean prior of each edge, i.e. $C_{ij} = 1$ if and only if $\sqrt{b_{ij} m_{ij}} \geq q$. Note that using the geometric mean of priors is equivalent to assuming $w_j, u_k = 0.5 \; \forall j, k$, and these parameters are initialized accordingly.

The threshold approach is generally appropriate for the first few runs of the estimation algorithm. For increased exploration of the solution space away from the most strongly predicted prior network, this initial deterministic starting network can be perturbed by randomly flipping the state of some fixed proportion of edges. For a more stochastic starting network, each edge parameter can be randomly sampled according to the geometric mean of its priors, i.e. $P\left(C_{ij} = 1\right) = \sqrt{b_{ij} m_{ij}}$.

Given a fixed network $(\mathbf{C}, \mathbf{D})$, the parameters $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ can be fit using the Ordinary Least Squares (OLS) solution to linear regression. To add a stochastic aspect to the selection of values for these parameters, the best-fit values can be perturbed by adding noise sampled from a normal distribution with $\mu = 0$ and small ν (i.e. 0.05). The model-wide variance is always initialized by $\sigma^2 = s^2$ (from Eq 1.17).

Phase 2: Local Maximization

Given initial values for all model parameters, it is now possible to compute the individual posterior of any model variable according to the equations in Sec 1.3. A hill-climbing algorithm is run from the starting point, such that each parameter is reassigned by maximum likelihood as derived from the equations above:

- Set each $\alpha_i = \mu_{\alpha_i}$ (from Eq 1.8)
- Set each $\beta_j = \mu_{\beta_j}$ (from Eq 1.11)
- Set each $\gamma_{jk} = \mu_{\gamma_{jk}}$ (from Eq 1.14)
- Set $\sigma^2 = s^2$ (from Eq 1.17)
- For each C_{ij}:
  - Compute $Z_0^{(C_{ij})}, Z_1^{(C_{ij})}$ (from Eq 1.19, 1.20)
  - Set $C_{ij} = 1$ iff $Z_1^{(C_{ij})} > Z_0^{(C_{ij})}$, otherwise set $C_{ij} = 0$
- For each D_{jk}:
  - Compute $\zeta_0^{(D_{jk})}, \zeta_1^{(D_{jk})}$ (from Eq 1.21, 1.22)
  - Set $D_{jk} = 1$ iff $\zeta_1^{(D_{jk})} > \zeta_0^{(D_{jk})}$, otherwise set $D_{jk} = 0$
- Set each $w_j = \arg\max_\lambda \left(p_\lambda\right)$ (from Eq 1.23).

- Set each $u_k = \arg\max_\lambda (p_\lambda)$ (from Eq 1.24).
- Repeat until convergence

The hill-climbing algorithm is run until the model parameters converge on an optimal solution. Convergence is determined by tracking the log of the model likelihood (rather than the model posterior, because the parameter priors are constant), given by Eq 1.25 below. Convergence is assumed when $\Delta \ln(L) < 0.0001$ (approximately 0).

$$\begin{aligned}\ln(L) =& \frac{-NT}{2} \ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon_{it}^2 \\ &+ \sum_{i=1}^{N} \sum_{j=1}^{J} w_j \left[C_{ij} \ln(b_{ij}) + (1 - C_{ij}) \ln(1 - b_{ij})\right] + (1 - w_j) \left[C_{ij} \ln(m_{ij}) + (1 - C_{ij}) \ln(1 - m_{ij})\right] \\ &+ \sum_{j=1}^{J} \sum_{k=1}^{K} u_k \left[D_{jk} \ln(a_{jk}) + (1 - D_{jk}) \ln(1 - a_{jk})\right] + (1 - u_k) \left[D_{jk} \ln(s_{jk}) + (1 - D_{jk}) \ln(1 - s_{jk})\right]\end{aligned} \tag{1.25}$$

Phase 3: Local Network Estimation

The final phase of the algorithm collects local samples of specific variables of interest, notably $\mathbf{C}$ and $\mathbf{D}$, while leaving other parameters, notably $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, fixed at the values identified in phase 2. This results in a single estimate for each parameter in $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and fixing these variables helps to anchor the subsequent sampling iterations around the local optimum. The $\boldsymbol{\gamma}$ parameters must also be resampled in this phase, to allow for informative estimation of $\mathbf{D}$. Specifically, if a parameter $\gamma_{jk} = 0$ at the end of phase 2, and if this value remains fixed, then the posterior for D_{jk} becomes equivalent to its prior, and therefore uninformative. The variables σ^2, $\mathbf{w}$, and $\mathbf{u}$ are also resampled, to allow for more robust resampling of $\mathbf{C}$ and $\mathbf{D}$. These variables are resampled in the same order as they were maximized in phase 2, using sampling rules derived from the equations above. This phase of the algorithm is repeated a fixed number of times:

- Sample each $\gamma_{jk} \sim \mathrm{N}(\mu_{\gamma_{jk}}, \nu_{\gamma_{jk}})$ (from Eq 1.15, 1.14)
- Sample σ^2 from the scaled-inverse χ^2 distribution with $TN + 2$ degrees of freedom and scale parameter s^2 (from Eq 1.17)
- Sample each C_{ij} from the Bernoulli distribution with $p = \frac{Z_1^{(C_{ij})}}{Z_1^{(C_{ij})} + Z_0^{(C_{ij})}}$ (from Eq 1.19, 1.20)
- Sample each D_{jk} from the Bernoulli distribution with $p = \frac{\zeta_1^{(D_{jk})}}{\zeta_1^{(D_{jk})} + \zeta_0^{(D_{jk})}}$ (from Eq 1.21, 1.22)
- Sample each w_j using grid sampling as described for Eq 1.23

- Sample each u_k using grid sampling as described for Eq 1.24

The output of a single run of this algorithm is a set of locally optimal fixed values for parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, denoted from here on as $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\beta}}$, and a fixed number of samples for all other parameters. To avoid auto-correlation between the samples, a lag is introduced, such that samples are only used for model inference every ℓ iterations, and all other samples are dropped. From here on, let X denote the number of samples used for model inference, and let x denote a specific sample from 1 to X. In other words, phase 3 is run for a total of $\ell \times X$ iterations, but only X samples (every ℓ^{th} iteration) are used for model inference.

1.5 Model Inference

Using a single run of the algorithm outlined above, we can obtain a maximum *a posteriori* (MAP) value for each parameter α_i and β_j from $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\beta}}$ fixed at the end of phase 2. Subsequently, we can compute partial posteriors for all other parameters, given both the input data, and these MAP values. The partial posterior for the remaining model parameters is:

$$\hat{P}(\mathbf{C}, \mathbf{D}, \boldsymbol{\gamma}, \sigma^2, \mathbf{w}, \mathbf{u} | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}) \tag{1.26}$$

However, we are generally more interested in computing this partial posterior for each parameter, marginalized over all other parameters. For example, to estimate the probability of a particular TF-Gene edge C_{ij}, we would need to compute $\hat{P}(C_{ij} | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}})$. The marginalized partial posteriors are computed from the X samples computed in phase 3, with $[x]$ denoting the value of a particular parameter in sample x, as follows:

$$\hat{P}\left(C_{ij} = 1 | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = I\left(\hat{\beta}_j \neq 0\right) \frac{1}{X} \sum_{x=1}^{X} C_{ij}[x] \tag{1.27}$$

$$\hat{P}\left(D_{jk} = 1 | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \frac{1}{X} \sum_{x=1}^{X} I\left(\gamma_{jk}[x] \neq 0\right) D_{jk}[x] \tag{1.28}$$

$$\hat{E}\left(\gamma_{jk} | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \frac{1}{X} \sum_{x=1}^{X} I\left(D_{jk} = 1\right) \gamma_{jk}[x] \tag{1.29}$$

$$\hat{E}\left(\sigma^2 | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \frac{1}{X} \sum_{x=1}^{X} \sigma^2[x] \tag{1.30}$$

$$\hat{E}\left(w_j | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \frac{1}{X} \sum_{x=1}^{X} w_j[x] \tag{1.31}$$

$$\hat{E}\left(u_k | \Xi, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\right) = \frac{1}{X} \sum_{x=1}^{X} u_k[x] \tag{1.32}$$

The function I denotes the identity function, and returns 1 if the specified condition is true, or 0 otherwise. Eq 1.27 corresponds to the probability that TF j regulates gene i. Eq 1.28 corresponds to the probability that modifier k has a synergistic effect on the activity of TF j. Eq 1.29 corresponds to the expected value of γ_{jk} for a particular synergistic effect, and the magnitude of this value can be used as an additional filter to select only those TF-Modifier edges with substantial effects on target gene expression. In this work we typically use a threshold of $|\gamma_{jk}| \geq 0.05$ to rule out TF-Modifier connections with inconsequential effects on target gene expression. In general, $\beta_j = 0$ (TF j has no influence on target genes) is considered equivalent to $C_{ij} = 0\ \forall i$ (TF j has no target genes), and $\gamma_{jk} = 0$ (TF j and modifier k do not have a synergistic effect) is considered equivalent to $D_{jk} = 0$ (TF j and modifier k do not interact). These equivalencies are built into the inference calculations above. Eq 1.31 is used to estimate the relative quality of priors for each TF j, with values > 0.5 favoring $\mathbf{b}$ and values < 0.5 favoring $\mathbf{m}$. Likewise, Eq 1.32 estimates the relative quality of priors for each modifier k, with values > 0.5 favoring $\mathbf{a}$ and values < 0.5 favoring $\mathbf{s}$. Thus, given a compendium of reliable experimental data as input, the parameters of this model can be fit to the data in order to make inferences about the underlying biological network.

Multiple Result Summarization

Running the three-phase algorithm multiple times is likely to give varying solutions to the fitting problem, especially when each run is initiated with a semi-random set of parameters. While the results of individual chains may be of interest, there is a need to summarize across multiple results in order to score each edge in terms of how likely it is *overall* given the results of multiple runs. We do this by simply averaging together the expected or fixed value of each parameter in Θ and averaging together the estimated posterior probabilities for each edge variable in $(\mathbf{C}, \mathbf{D})$.

The averaging method is only valid under the assumption that all individual result sets are similar - i.e. each is an estimate around the same mode of the solution space. To assess the validity of this assumption, we perform hierarchical clustering [3] of the individual network models returned by each run of the algorithm. Multi-modal posteriors typically arise in our model when several disparate networks can explain the available data. Therefore, we use the vector of posterior probabilities assigned to all edge variables $(\mathbf{C}, \mathbf{D})$ to represent each model and then compute the Euclidean distance between all pairs of models. From this distance matrix we perform complete hierarchical clustering as implemented in R [4] and visually inspect the resulting tree. If the overall tree structure does not show any clearly delineated clusters, then the computed network models most likely represent estimates of the same solution mode and can be averaged together for a more robust estimate. The clustering results from our Yeast application exemplify this criteria (see Section 4).

Conversely, if the tree shows several well-separated clusters, then the computed network models were most likely sampled from multiple divergent solution modes. In this case, we select an appropriate height cut-off for clustering the models, and then average together and perform biological inference separately within each cluster. The clustering results from our

human B cell application appear to be multi-modal based on this criteria (see Section 5). At this time we do not have a solution for summarizing over multiple modes, and prefer instead to analyze the biological significance of each cluster individually. It is possible that the underlying or "true" network that produced the input data is non-constant, possibly due to a mixture of cell types or genetic backgrounds included in the input data, or due to biological behavior not captured by the model. In this case each mode may capture a different static view of the dynamic network. The phenomenon of multi-modality can also occur due to violations of model assumptions, e.g. a TF acting as activator for one set of genes and repressor for another. Thus, considering multiple modes in the parameter space allows us to overcome these model limitations and identify both activated and repressed TF targets.

2 Data Sources and Pre-Processing

In this section, we describe all procedures used to prepare input data for the MONSTER applications. We begin with several general rules relevant to all applications, and then cover specific details for each application.

When defining the expression matrices, the genes used as transcription factors (TFs) and modifiers for a particular model application are always completely excluded from the set of target genes. This is necessary because we use the expression data as a proxy for activity of these regulators and attempting to simultaneously model their transcriptional regulation would more likely result in false positive connections. However, this limitation could be overcome if both protein-level and transcript-level measurements were available for TFs and modifiers in all conditions.

In order to derive PWM-based priors of any kind for TF-Gene or TF-Modifier edges), each PWM is scanned along a DNA or protein sequence to compute a score at each possible PWM-sequence alignment. Each score can be converted to a p-value by comparing it to a distribution of scores computed for some set of background sequences (e.g. random genomic or protein sequences). This procedure is described in more detail by Levy, et al. [5], and the specific background sequences used are noted in each appropriate section. A conversion from p-values to prior probabilities was first introduced by Chen, et al. [1], and is reviewed here where applicable (Eq 2.2 and 2.3).

As noted by Chen, et al. [1], all priors (PWM-based or otherwise), must be "trimmed" so that they do not include extreme values of 0 or 1, as these prevent further estimation of posteriors by our model. We limit the range of all priors to $(0.05, 0.95)$ for all applications. The distributions of all priors are shown in Figures S3 and S4, and explained in more detail below.

2.1 Simulated Data

Simulated data was generated according to model equations (Sec 1.1) in order to test the accuracy of the model-fitting method (Sec 1.4). The initial simulated network had $N = 200$

target genes, $J = 10$ TFs, $K = 100$ modifiers, and $T = 100$ conditions. The expression values for all TFs and modifiers in all conditions ($\mathbf{f}$ and $\mathbf{h}$) were randomly sampled from the normal distribution $\mathrm{N}(0, 1)$. The edge indicator variables C_{ij} and D_{jk} were selected at random with the following constraints:

- Each TF regulates at least 50 and at most 100 genes: $50 \leq \left[\sum_{i=1}^{N} C_{ij}\right] \leq 100 \ \ \forall j$
- Each TF is targeted by at least 1 and at most 20 modifiers: $1 \leq \left[\sum_{k=1}^{K} D_{jk}\right] \leq 20 \ \ \forall j$
- Each modifier targets at least 1 and at most 3 TFs: $1 \leq \left[\sum_{j=1}^{J} D_{jk}\right] \leq 3 \ \ \forall k$

Each parameter β_j was randomly sampled from the normal distribution $\mathrm{N}(0, 0.3)$, but was resampled whenever $|\beta_j| < 0.05$. Each parameter γ_{jk} for which the corresponding $D_{jk} = 1$ was randomly sampled following the same procedure used for the β_j variables. The prior matrices ($\mathbf{b}, \mathbf{m}, \mathbf{a}, \mathbf{s}$) were all randomly sampled using the standard Beta distribution to generate priors that are correlated to the intended network, but also contain a substantial amount of noise. Each prior value was sampled as follows:

$$
\begin{aligned}
b_{ij}, m_{ij} &\sim \mathrm{B}(z, 1) && \forall (i,j) : C_{ij} = 1 \\
b_{ij}, m_{ij} &\sim \mathrm{B}(1, z) && \forall (i,j) : C_{ij} = 0 \\
a_{jk}, s_{jk} &\sim \mathrm{B}(z, 1) && \forall (j,k) : D_{jk} = 1 \\
a_{jk}, s_{jk} &\sim \mathrm{B}(1, z) && \forall (j,k) : D_{jk} = 0
\end{aligned}
$$

Note that in the equations above, B denotes a standard probability distribution (the "Beta" distribution) and should not be confused with our prior matrix $\mathbf{b}$ or the model parameters $\boldsymbol{\beta}$. The variable z is randomly sampled from the uniform distribution $\mathrm{U}(1.2, 1.5)$ independently for each column of each matrix, in order to create varying prior "quality" for each TF and each modifier.

The expression values for all genes in all conditions ($\mathbf{g}$) were computed according to Eq 1.1 with all residual error terms ε_{it} initially set to 0. Additional versions of $\mathbf{g}$ were calculated with residual errors randomly sampled according to Eq 1.2 using increasing values of $\sigma^2 = 0.05, 0.1, \ldots, 0.95, 1$, denoted $\mathbf{g}_{\sigma^2=0.05}$, etc.

For subsequent simulations, the input data was expanded to include either additional, uninformative target genes, or additional, uninformative conditions. In both cases, the matrix $\mathbf{g}_{\sigma^2=1}$ was used as a starting point, so these subsequent simulations also include a considerable amount of noise in the informative target genes and conditions.

To add additional, uninformative target genes, the model was expanded to include $N = N_0 + N'$ target genes, where $N_0 = 200$ for the original, informative target genes, and $N' = 20, 40, \ldots, 200$ for the additional, uninformative genes. All uninformative expression values randomly sampled from $\mathrm{N}(0, 1)$. In other words, each additional gene profile was completely random, with no information related to the network structure.

To add additional, uninformative conditions, the model was expanded to include $T = T_0 + T'$ conditions, with $T_0 = 100$ for the original, informative conditions, and $T' = 10, 20, \ldots, 100$

for the additional, uninformative conditions. All expression values in the uninformative conditions were sampled from the distribution $\mathrm{N}(0, 1)$. In other words, each additional condition contained completely random expression values, with no information related to network structure.

2.2 Yeast Application

Yeast Expression Compendium

In the yeast application, we derived the expression matrices $\mathbf{g}$, $\mathbf{f}$, and $\mathbf{h}$ from a compendium of $T = 314$ microarray samples, previously compiled and normalized by Chen, et al. [1]. The matrix $\mathbf{f}$ contains expression profiles for the TFs MSN2 (YMR037C) and MSN4 (YKL062W). Table S3 lists the 40 known targets of MSN2/4 and 40 decoy targets contained in matrix $\mathbf{g}$, and the 81 kinases contained in matrix $\mathbf{h}$.

An additional "z-score" normalization was applied to expression profiles for each gene, TF, and modifier. This step is necessary to strengthen the assumption of uniform model-wide error variance (Eq 1.2). This normalization also strengthens our assumption that the prior distribution for each α is centered at 0, and helps remove differences in mean and variance from skewing the β and γ parameters. Thus, the normalized values can be thought of as expression relative to the normal range observed for each particular gene in the conditions tested, and our regression parameters are therefore also on a scale of relative influence. The normalization is shown for target genes in Eq 2.1, although the same process was also applied to values of f_{jt} and h_{kt} accordingly:

$$g_{it} = \frac{g_{it}^{(\mathrm{raw})} - \mu_{g_i}^{(\mathrm{raw})}}{\sigma_{g_i}^{(\mathrm{raw})}} \quad \forall i, t \tag{2.1}$$

MSN2/4-Target Gene Priors from PWMs

In the yeast application, we used a single matrix $\mathbf{m}$ for TF-Gene priors. To derive this matrix, we used PWMs for MSN2 and MSN4 previously computed by Harbison, et al. [6]. For each TF-Gene pair (i, j), we scanned both strands of the 700bp upstream region for each gene i, and recorded the best (maximum) PWM score, denoted as S_{ij}. We then scanned the same PWMs against the promoter regions of all ($\sim$6000) yeast genes to generate a background score distribution for each PWM, thereby allowing us to convert each score S_{ij} to a p-value P_{ij} [5]. We then converted each p-value to a prior probability following the same equation used by Chen, et al. [1]:

$$m_{ij} = (1 - P_{ij})^{2[\ell - W_j + 1]} \tag{2.2}$$

In this case, $\ell = 700$ is the fixed promoter length, and W_j is the width (in bases) of the PWM corresponding to TF j. The distribution of these priors is shown in Figure S3.

Kinase-Substrate Priors from PWMs

We generated a single matrix **s** of priors for TF-Kinase connectivity by scanning each TF protein sequence with a PWM describing the predicted substrate specificity of each kinase. We first acquired the protein sequence of each known kinase present in our expression compendium from the Saccharomyces Genome Database (SGD) [7] on 7/28/08. We then submitted each sequence to the Predikin v2.0 web server [8].

Briefly, Predikin takes a kinase protein sequence as input, and constructs a predicted substrate profile, as a PWM, from the known substrates of related kinases. See [8] for further details. Thus, for each kinase protein sequence, Predikin returns a predicted PWM describing the most likely protein sequences to be targeted by that kinase. Kinases for which the Predikin server failed were discarded from our input set.

Next, we obtained the protein sequences for the TFs MSN2 and MSN4, also from the SGD [7] on 7/28/08. We scanned the PWM for each kinase k against the protein sequence for each TF j, and recorded the maximum PWM score, denoted as S_{jk}. We also generated and scanned scrambled versions of these same protein sequences to create an appropriate background score distribution for each PWM. For each kinase k, we used the background distribution of PWM scores to transform all corresponding scores S_{jk} to p-values, denoted P_{jk}. We then convert p-values to prior probabilities, adapting Eq 2.2 above:

$$s_{jk} = (1 - P_{jk})^{\ell_j - W_k + 1} \tag{2.3}$$

In this case, ℓ_j is the length of the protein sequence for TF j, and $W_k = 7$ for all PWMs generated by Predikin [8]. The distribution of these priors is shown in Figure S3.

Missing Priors: ChIP-chip and Kinase-Substrate Interaction Data

Note that our yeast input data set lacks TF-Gene priors **b** based on ChIP experiments. While there is publicly available genome-wide ChIP-chip data for our factors of interest, these experiments were performed in the absence of any environmental stress [9]. As noted in our manuscript, MSN2/4 are primarily controlled at the level of nuclear transport, and in the absence of stress, these factors are not present at high concentrations in the nucleus. Unsurprisingly, few binding sites were identified for these particular factors by this method. We attempted to derive prior probabilities **b** from this data set, but found that they resulted in an overly sparse network with lower performance (data not shown).

There is also publicly available data for yeast kinase substrates identified *in vitro* through the use of protein microarrays [10]. We attempted to use this data set to derive priors **a**, but ultimately rejected it for several reasons. For one, quantitative data was not made public, and so we can only assign a "high" prior (i.e. 0.9) to their predicted kinase-substrate connections, and a "low" prior (i.e. 0.1) to all other connections. This data set was also especially sparse for the TFs MSN2 and MSN4, and in particular no kinases were predicted to phosphorylate MSN2. We also attempted to derive priors from STRING [11], using the same method as in our human B cell application (see Section 2.3). Again, this resulted in an overly sparse network with lower performance (data not shown).

2.3 Human B Cell Application

Human B Cell Expression Compendium

In the human B cell application, we derived the expression matrices $\mathbf{g}, \mathbf{f}, \mathbf{h}$ from a compendium of $T = 336$ microarrays available in GEO [12] (accession GSE2350), and previously published by Basso, et al. [13]. We processed the raw data using RMA [14] in BioConductor [4], which outputs normalized expression values on the $\log_2$-scale. We then filtered out all probe sets with no expression in any condition (max expression $< \log_2(100)$), or insufficient perturbation across the samples (variance < 0.03). All remaining probe sets were further normalized across all samples by subtracting the mean expression value and dividing by the standard deviation of the profile, as performed for the yeast expression data (Eq 2.1).

The expression data contained 4 different probe sets corresponding to STAT1, which were highly correlated ($r^2 > 0.9$ for all pairs). We calculated the mean of these expression values within each sample to create a single representative expression profile for STAT1 ($\mathbf{f}, J = 1$). We also identified groups of probe sets that corresponded to the same kinase or phosphatase gene and combined redundant expression profiles. For each Entrez Gene ID [15] corresponding to multiple probe sets in $\mathbf{h}$, we repeated the following procedure:

1. Identify all corresponding probe sets (expression profiles)
2. Identify the pair of expression profiles with the largest positive correlation, r^2
3. If $r^2 > 0.5$, average the corresponding pair of expression values within each sample, and replace both profiles with the average expression profile
4. Repeat steps 2 and 3 until the criteria in step 3 is not met, or until a single expression profile remains.

We combined probe sets for TFs and modifiers in our model because redundant regulators could lead to extraneous network connections, in which the duplicates of the regulator expression profile cancel each other out in the regression model. However, we still allow for a modifier to be represented by multiple expression profiles in our model input data if the probe sets are not strongly correlated across the expression compendium. This process resulted in $K = 510$ expression profiles for kinases and phosphatases ($\mathbf{h}$), corresponding to 323 unique Entrez Gene IDs. The remaining $N = 8,973$ probe sets that passed the initial filtering, but did not correspond to STAT1, kinase, or phosphatase genes, were treated as target genes ($\mathbf{g}$). No clustering was applied to target gene probe sets, regardless of correlation, and redundant target gene profiles have no impact on other model parameters (data not shown). Thus, the probe sets correspond to only $7,026$ unique Entrez Gene IDs. Combined expression profiles for STAT1 and relevant modifiers were z-score normalized by the same method as target genes (Eq 2.1).

STAT1-Target Gene Priors from ChIP-seq

We obtained ChIP-Seq data for STAT1 from GEO [12] (accession GSE12782), previously published by Rozowsky, et al. [16]. This data includes reads for STAT1 ChIP-seq experiments

and input DNA controls from IFN-γ-treated HeLa S3 cells. We first filtered this data by removing all ambiguous, unmapped, and redundant reads, then applied the GLITR algorithm for peak-calling, as described in [17]. Here, we briefly summarize the GLITR algorithm, with emphasis on the steps adapted for our application.

GLITR compares the ChIP-seq reads to input DNA reads in two different ways. At each potentially enriched region, GLITR directly compares the ChIP-seq peak height to random samples from a substantially larger pool of "background" input reads, in order to compute the median fold change of each peak. GLITR also defines a set of "Pseudo" reads, which are also compared to the background reads in the same way as the ChIP-seq reads. These Pseudo peaks are then compared globally to the GLITR peaks using a nearest neighbor method based on both absolute peak height and median fold change against background. This global comparison produces the FDR thresholds ultimately used to select a set of high-confidence ChIP peaks. ChIP peaks which directly overlap a Pseudo peak at this threshold are also removed.

As discussed in [17], the Pseudo set can either be a paired control experiment, or can be sampled out of the background set. The background set itself can come from a paired control experiment, but only if the control has a much larger sequence depth than the ChIP sample, which is uncommon. Therefore, Tuteja et al. developed a pool of background reads from multiple input samples that is appropriate for most cell types [17]. In the case of the STAT1 data, the paired control resulted in approximately the same number of reads as the ChIP-seq, and therefore we used the filtered input DNA controls from [16] as "Pseudo" reads, and all input human DNA reads from [17] as "background". We randomly removed a small percentage ($< 0.1\%$) from the remaining ChIP-seq start coordinates, such that both the ChIP-seq and Pseudo input sets had the same number of unique start coordinates, as required by the GLITR algorithm. All other GLITR parameters were left at their default settings.

GLITR outputs a score $X \in \{0, 1, \ldots, 100\}$ for each ChIP and Pseudo peak, with 0 corresponding to the most confident peaks. The proportion of Pseudo regions passing a given threshold $X \leq x$ is used as an estimate of the false positive rate among the ChIP regions passing this same threshold, based on the assumption that Pseudo data provides a suitable model of randomly occurring input peaks. Thus, the FDR at any threshold x is given by Eq 2.4, originally from [17]:

$$\text{FDR}(x) = \frac{\text{Proportion of Pseudo peaks with } X \leq x}{\text{Proportion of ChIP peaks with } X \leq x} \tag{2.4}$$

For our purposes, we need to map each peak score X to a prior probability, rather than use an FDR cutoff. To map scores X to prior probabilities p_X, we rely on the same assumption as the FDR calculation, namely that the distribution of scores for the Pseudo peaks is an estimate of the false positives among the ChIP peaks. Thus, we can derive a reasonable estimate of the desired probability

$$p_x = 1 - \left(\frac{\text{Proportion of Pseudo peaks with } X = x}{\text{Proportion of ChIP peaks with } X = x} \right) \tag{2.5}$$

Note that this differs from the FDR estimate in that it uses the proportions of regions at a precise score, rather than passing a threshold. In practice, the number of regions are under-sampled for some scores, and therefore the approximation of p_x is not robust for all x. To adjust for this problem, we took two basic steps. First, we binned the scores X into bins of size 5, and computed a smoothed probability for the entire bin. In other words, $p_0 = \cdots = p_4 = p_{X \in (0,4)}$, and so on. The majority of peaks have $X = 100$, and so p_{100} was computed as a single bin. Second, we assumed that a correct mapping $X \rightarrow p_X$ should be monotonically decreasing. In other words, if score x_1 is better (lower) than another score x_2, then $p_{x_1} \geq p_{x_2}$ in all such cases. We enforce the monotonicity by simply iterating through x from 1..100, and assign $p_x = \max(p_x, p_{x-1})$. The final mapping from $X \rightarrow p_X$ is shown in Figure S5. Note that it is desirable that $p_{100} = 0$, as these are the majority of the false positives in the data, and regions with $p = 0$ can be ignored for subsequent analysis. In this application, $p_x = 0 \ \forall x > 64$, effectively filtering out most poor-scoring peaks in the data.

The GLITR algorithm does not explicitly compare Pseudo and ChIP peaks that overlap in the genome. Such cases may be indicative of regions where DNA becomes more accessible as a result of IFN-γ treatment, but where there is no actual STAT1 binding site. To account for these regions, we applied the same mapping $X \rightarrow p_X$, derived for ChIP peaks, to the scores for all Pseudo peaks. Let p_c be the probability of a particular ChIP peak, and let p_u be the probability of some overlapping Pseudo peak. Let p_f denote the final probability assigned to the ChIP region after consideration of overlapping Pseudo regions, calculated by:

$$p_f = \max(0, p_c - \max p_u) \tag{2.6}$$

In other words, simply subtract the probability of the strongest overlapping Pseudo peak from the probability of the ChIP peak. This essentially enforces a "penalty" on ChIP peaks with overlapping Pseudo regions. If the Pseudo peak has probability greater than or equal to that of the ChIP peak (if $p_u \geq p_c$) then the ChIP peak is removed entirely ($p_f = 0$).

The goal of this analysis is to derive prior probabilities for the regulation of target genes by STAT1, rather than for binding to genomic regions in general. Therefore, we must also map individual peaks with $p_f > 0$ to proximal genes in order to derive final values for $\mathbf{b}$. The most reasonable and straightforward way to perform this mapping is to simply define a presumed promoter region as a fixed amount of sequence upstream of each gene start site, then take the maximum p_f for all peaks overlapping the presumed promoter. In this analysis, we defined the presumed promoter as the 1kb region upstream of each gene start site as defined in RefSeq [18], in order to be consistent with the PWM-based priors (below). Thus, for each gene i, and j =STAT1, $b_{ij} = \max p_f$ using all peaks overlapping the designated 1kb upstream region. The distribution of these priors is shown in Figure S4.

STAT1-Target Gene Priors from PWMs

STAT1 binds to two distinct motifs, depending on its dimerization partner and the upstream signal triggering its activity [19]. The IFN-Stimulated Response Element (ISRE) is typically bound by STAT1 in response to Type I IFNs (i.e. IFN-α/β), while the IFN-γ-Activated Site (GAS) is typically bound by STAT1 in response to the Type II IFN (IFN-γ). We used a

PWM from TRANSFAC [20] to represent the ISRE motif (accession M00258), and derived a PWM for GAS sites using 19 exemplary sequences compiled by Robertson, et al. [21] (Figure S6). These two PWMs were used in conjunction to derive the TF-Gene prior matrix **m** in the human B cell application.

We then scanned the same promoter regions used for the ChIP-seq analysis (1kb upstream) with both PWMs. For the background model to determine p-values, we used the 1kb upstream regions for all annotated human RefSeq genes [18], including those which could not be mapped to a transcript in our expression data. For each promoter i, we used the most significant p-value for either PWM, denoted P_{ij}. We then used an adjusted version of Eq 2.2 that accounts for the use of multiple PWMs to convert p-values to prior probabilities:

$$m_{ij} = (1 - P_{ij})^{2[2\ell-(W_I+W_G-2)]} \tag{2.7}$$

Here, $\ell = 1000$ is the length of the promoter sequence, W_I is the width of the ISRE PWM, and W_G is the width of the GAS PWM. The exponent computed in Eq 2.7 is the number of tests done against both PWMs for a single promoter (both strands of sequence are scanned). The distribution of these priors is shown in Figure S4.

Modifier-STAT1 Priors from STRING

We derived the matrix **s** using selected channels of interest from the STRING database [22]. We downloaded the detailed protein links (including individual channel scores) for STRING v8.2 on 10/23/09. We then mapped STAT1 and all modifiers in the expression matrix **h** to their corresponding STRING identifier via Entrez Gene Symbol [15], and extracted all protein links between STAT1 and any input modifiers. We recomputed the score S for each link using only the channel scores marked "`experimental`" (denoted here as S_e) and "`database`" (denoted here as S_d) using the Bayesian integration equation from [11]:

$$S = 1 - ((1 - S_e)(1 - S_d)) \tag{2.8}$$

Thus, each prior probability s_{jk} corresponds to the recomputed score S for the link between j = STAT1 and modifier k. All missing links were presumed to have $s_{jk} = 0$ (therefore transformed to 0.05 by our restricted prior range described above). The distribution of these priors is shown in Figure S4.

3 Analysis of Simulated Data

We analyzed simulated input data (Sec 2.1) in order to estimate the accuracy of our model-fitting method (Sec 1.4). For our first analysis, we ran our algorithm 10 times using as input the simulated expression matrices $\mathbf{g}_{\sigma^2=0}$, **f**, **h**, **b**, **m**, **a**, **s**, and seeded the algorithm by randomly selecting each network edge according to the geometric mean of its priors. This initialization procedure is fairly stochastic, representing wide coverage of the possible solution space, and is suitable for smaller networks. We then averaged together the model estimates from all 10 runs.

We then plotted a receiver operating characteristic (ROC) curve for the posterior probabilities **C** and **D** (Figure S1A,B). For comparison we also show ROC curves for each individual type of prior probability, and for TF-Gene expression correlations. The performance observed on the ROC curves can be quantified by computing the area under the curve (AUC) metric. We found that MONSTER perfectly recovered the TF-Gene edges (AUC = 1, solid blue line in Figure S1A). We evaluated the relative advantage of the full model, as compared with using either the priors **b** or **m** alone, or expression correlations between TFs and target genes, to predict the TF-Gene edges. We found that all alternatives fell short of using the full model, resulting in AUC values less than 0.75 (dotted and dashed blue lines in Figure S1). TF-Modifier edges were substantially harder to infer, owing to their indirect effect in the expression model (AUC = 0.78, solid green line in Figure S1B). However, MONSTER predicted these edges with better accuracy than either simulated prior type alone. These results demonstrate that the observed performance of MONSTER cannot be reproduced by simpler analyses of the individual data sources.

In subsequent trials, we increased various sources of noise in the simulated expression data (note that the priors are also noisy, but we did not vary this noise). For each analysis, we ran our algorithm 30 times, using the same stochastic initialization procedure as the previous analysis, and averaged the model estimates within groups of 10 runs, in order to also assess the variability of algorithm performance. For each group of 10 runs, we computed an average estimate of **C** and **D**, recomputed the ROC as in the first analysis, and computed the AUC. This resulted in 3 AUC values for each analysis (1 for each group of 10 runs), and we plotted these values as "I-bars" spanning the min and max AUC values, and a line drawn through the median AUC value for each analysis.

We performed this series of analyses by first gradually increasing the model-wide variance σ^2 from 0 to 1 (using matrices $\mathbf{g}_{\sigma^2=0}$ thru $\mathbf{g}_{\sigma^2=1}$ from Sec 2.1). Figure S1C shows that algorithm performance is unaffected by noise in the individual expression values, up to a degree of noise equivalent to the expression variance for each regulator (TF or modifier). Note that this limit is essentially guaranteed by the z-score transformation that we applied to real expression data.

For subsequent analyses, we kept the highest value of σ^2 used in the previous series, and added either additional gene profiles, or additional expression conditions, neither of which contain any information relevant to the underlying network. The motivation for this analysis is that real input data might contain (i) target genes that are not targeted by any of the TFs of interest, and (ii) expression samples in which the underlying network is disrupted due to non-model elements, i.e. samples from a different cell-type. As described in Sec 2.1, we added either N' uninformative genes, or T' uninformative conditions, with values ranging from 0 to N_0 or T_0 respectively. Figure S1D shows the results for increasing numbers of uninformative genes, represented as the percentage $(N'/N_0) \times 100$. Again, performance is essentially constant. Similarly, Figure S1E shows the results for increasing numbers of uninformative samples. In this case, the inference of TF-Modifier connections were negatively affected by this type of noise, but still out-performed individual prior sources.

Overall, we conclude from these results that our model-estimation method can operate

accurately in the presence of noise in both the expression and prior data, and is robust to the inclusion of uninformative gene profiles and expression samples.

4 Analysis of Yeast MSN2/4 Network

The yeast input data set $(\mathbf{g}, \mathbf{f}, \mathbf{h}, \mathbf{m}, \mathbf{s})$ was prepared as described in Sec 2.2. We applied our method by running the model-fitting method 100 times on the complete data set to assess overall algorithm performance, and another 100 times using only $\mathbf{g}$, $\mathbf{f}$, and $\mathbf{m}$ (TFs only, no kinases) to specifically assess the contribution of TF-Modifier synergy terms to target gene expression prediction. In each run, we seeded the algorithm using the same stochastic method as for the simulated data analysis (note that the network sizes in both applications are comparable). We used a substantially larger number of runs here than for the simulated data because: (1) we were performing fewer individual analyses, and could dedicate more run time to each set of input data; (2) we observed a clear increase in accuracy (AUC) using a larger number of runs in this case (data not shown).

We performed hierarchical clustering on the 100 individual network models estimated as described in Sec 1.5, for both the full model, and the TF-only model (Figure S7). In both cases, the major clusters are not well-separated (most of the tree height is within these clusters). This meets our criteria for a single solution mode, although there is clearly some variability between individual model estimates.

We computed the accuracy of each model based on the known targets and modulating kinases for MSN2/4, as described in the main manuscript. In order to assess the significance of the ROC curves in Figure 3 (main text), we performed two additional types of analysis: permutation tests and bootstrapping.

Permutation tests [23] were performed for each individual ROC curve, to determine the significance of the computed AUC values. In this test, a set of edge discriminants, i.e. the posterior estimates $\mathbf{C}$, are randomly permuted relative to the known true/false labels on these same edges. The AUC value is recomputed for each permutation, and the p-value is the proportion of permutations in which the AUC value was higher than the originally observed AUC value. We performed 100,000 permutations for the AUC values corresponding to the following discriminants: full model posteriors $\mathbf{C}^{(+)}$, TF-only model posteriors $\mathbf{C}^{(-)}$, PWM-based priors $\mathbf{m}$, magnitude of expression correlation between $\mathbf{g}$ and $\mathbf{f}$, and global mutual information (MI) estimated by MINDY [24]. The resulting p-values are summarized in Table S4 (3^{rd} column). All AUC values were significant according to this test, indicating that each discriminant performs significantly better than random selection of edges. In other words, none of the AUC values in Table S4 are expected to occur by chance.

We also wanted to examine whether the apparent improvement in AUC observed for our full model posteriors $\mathbf{C}^{(+)}$ was truly significant, or might be expected to occur by chance. To test this, we performed a bootstrap analysis [23] as follows. For each of 100,000 bootstrap samples, we randomly sampled 160 TF-Gene edges from our actual network *with replacement* and with all assigned posteriors, priors, etc. Thus, for each individual bootstrap sample, some edges may be ignored, and other edges may count multiple times, although the overall

sample size remains constant. If the observed improvement in AUC is based on a small percentage of the overall network, then this apparent improvement should disappear in many of the bootstrap samples. Thus, the p-value based on bootstrap analysis is the proportion of bootstrap samples in which the AUC for the full model, $\mathbf{C}^{(+)}$, is no longer greater than another comparative AUC value (i.e. $\mathbf{C}^{(-)}$, the posteriors from the TF-only model). These p-values are reported for each of the AUC values as compared to the full model AUC (Table S4, 4^{th} column). For each bootstrap sample, we also computed the exact different in AUC values between $\mathbf{C}^{(+)}$ and each other discriminant, and summarized these values as a 95% confidence interval (Table S4, 5^{th} column). All bootstrap p-values were significant, and all differences in AUC had positive confidence intervals, indicating that the apparent improvement in accuracy observed in our full model vs. all other tested discriminants is unlikely to occur by chance. Most notably, this result indicates that the inclusion of kinases in the model fitting procedure resulted in a significant improvement to the overall accuracy of MSN2/4 target selection, as compared to the "TF-only" model (bootstrap p-value = 0.0018). Overall, the inclusion of modifiers in MONSTER improves our ability to model MSN2/4 regulation of target genes, and enables the prediction of upstream components for the transcriptional program.

We also performed 5-fold cross-validation in order to assess the possibility of over-fitting our model to the available data. We randomly divided the 314 conditions into 5 equal sized partitions, and repeated our entire model-fitting procedure for both the full and TF-only models with each partition held out (training the model on 80% of the available data). We then assessed the degree to which the network model could predict the observed gene expression values in the held out data by computing the mean square of all residual error values ε_{it} (MSRE) in the held out conditions t, and also computed this value for the training data for comparison. The mean and standard deviation of the MSRE on both the test and training data for both models are shown in Figure S8. Although the primary goal of our model is not the prediction of gene expression patterns, but rather the inference of the underlying network, these results show that our full model performs equally well on both the test and training data. Therefore, our full model does not show any indication of over-fitting despite the larger number of parameters relative to the TF-only model.

5 Analysis of the STAT1 Network in Human B Cells

The human B cell data we prepared in Sec 2.3 is approximately 30-fold larger than the yeast network, and therefore each run of the model-fitting algorithm requires substantially more computing time. Specifically, the average computing times for a single run of our model-fitting algorithm in the yeast and STAT1 applications were 129s and 6,656s, respectively (roughly 50-fold increase in run-time). Rather than seed the algorithm with a highly stochastic network, as we did in the yeast application, we opted to use a smaller number of runs seeded with initial models closer to the network predicted by the priors. We began by seeding our model-fitting method with a fixed network containing only those edges with high prior probabilities. We ran the method 3 times with a fixed start, in order to capture the small variation arising in the estimation phase. We then added a moderate amount of

perturbation to the starting model, in order to explore a larger portion of the solution space. We perturbed the network structure by adding/removing edges at random. Alternatively, we perturbed the starting values of numerical parameters by adding noise sampled from a normal distribution. Thus, we ran the model-estimation heuristic multiple times, seeded as follows:

- Non-Stochastic (`NS`): All network edges initalized by: $C_{ij} = 1$ iff $\sqrt{b_{ij} m_{ij}} \geq 0.75$ and $D_{jk} = 1$ iff $a_{jk} \geq 0.75$, all other parameters fit by OLS (3 runs)
- Perturbed Network (`PN`): All network edges initalized as in `NS`, then 1% of edges switched at random, all other parameters fit by OLS (10 runs)
- Perturbed Parameters (`PP`): All network edges initalized as in `NS`, all other parameters fit by OLS, then added random noise sampled from $\mathrm{N}(0, 0.05)$ (10 runs)

The hierarchical clustering dendrogram for the 23 model estimation runs is shown in Figure S9. Note that there are much longer branches at the top of the hierarchy, indicating well-separated clusters. Runs are labeled by the type of seeding method described above, and the three major clusters are outlined in green, purple, and blue and numbered I–III. We averaged model estimates together within each of these clusters to produce three distinct network models. Further perturbations to the initialization phase did not identify any additional solution modes (data not shown).

As an overall comparison of the three network models, we first looked at basic model properties, summarized in Table S9. The term β_1 is the influence parameter for STAT1, and describes the influence of STAT1, in general (independent of modifiers), and whether it is an activator or repressor. The term w_1 is the prior weight parameter for STAT1, and describes whether the inferred network is more dependent on the ChIP-based priors $\mathbf{b}$, or PWM-based priors $\mathbf{m}$. The number of targets is given in terms of probe sets, and is based on the threshold $P(C_{ij} = 1) \geq 0.9$. There is little overlap in the target genes predicted by each model. Models I and III share only 5 predicted targets, and clusters II and III share 95 predicted targets. Clusters I and II do not share any predicted targets. This supports the notion that all three clusters of model estimates are well-separated.

Cluster I produces the most reliable model for several reasons. Most notably, the predicted targets in this model are significantly enriched for known direct targets of STAT1, previously compiled by Roberston et al. [21]. 23 of these known targets mapped to 39 probe sets in matrix $\mathbf{g}$ and of these, 25 (64%) were predicted by MONSTER (hypergeometric test p-value = $2.6\text{x}10^{-9}$). The 23 known direct target genes are listed in Table S5 with the corresponding HGU95A probe set IDs present in the human B cell expression compendium.

Model I is also the only model with $\beta_1 > 0$, which is in agreement with the established role of STAT1 as an activator of transcription [25, 26]. This model has w_1 closest to 0.5, which indicates balanced use of the ChIP and PWM-based TF-Gene priors. By comparison, model II is biased somewhat towards the ChIP-based priors, which are limited to a singular cellular condition. Model III is heavily biased towards the PWM-based priors, which are

likely to have lower specificity in general compared to the ChIP-based priors. Model III also predicts that STAT1 has very weak influence on the predicted target genes ($\beta_1 \approx 0$).

To further compare these network models, we plotted histograms of the TF-Gene priors, expression correlations to STAT1, and expression entropy [27] for the target genes predicted by each network (Figure S10). These distributions further support the selection of model I as the most interesting and reliable solution mode. Despite the fact that model II (purple bars) is more heavily weighted to use ChIP-seq priors $\mathbf{b}$, and is generally more enriched for target genes with higher b_{ij} priors, model I (green bars) is actually the most enriched for the group of target genes with the highest b_{ij} scores (Figure S10A). This is also true for the PWM priors $\mathbf{m}$ (Figure S10B). In other words, model I is the most enriched for target genes with the strongest prior scores based on both ChIP-seq and PWM. Model I is also enriched for genes with the strongest correlation to STAT1 expression (Figure S10C).

Entropy [27] measures the overall broadness or lack of condition-specificity for each target gene. Lower entropy indicates that a gene is specifically expressed in a smaller set of expression samples in our compendium. Gene expression profiles with low entropy are the most likely to represent active regulatory patterns rather than random noise. Model III (blue bars), which also contains the largest number of predicted targets, and the weakest STAT1 regulatory influence (Table S9), is heavily biased towards target genes which appear to be the most random in their expression patterns (Figure S10D). Model I is highly enriched for genes with lower entropy, and therefore represents the strongest regulatory signal in the input data.

Based on the model comparisons described above, we chose model I as our "primary" model, i.e. the model most likely to capture the direct regulatory effects of STAT1. This model is discussed in detail in the main manuscript, and further details are described in the next subsection. The following subsections apply similar analysis procedures to the alternate networks and discuss the possible biological implications of these network models. For all models, we used the thresholds $P(C_{ij} = 1) \geq 0.9$ to predict high-confidence STAT1 target genes, and $P(D_{jk} = 1) \geq 0.9$, $\gamma_{jk} \geq 0.05$ to predict high-confidence STAT1 modifying enzymes.

5.1 Primary STAT1-Mediated Network Model

The primary model predicts 1,803 probe sets mapping to 1,559 unique genes as STAT1 targets. The details of these putative target genes are discussed in the main manuscript. The primary model also predicts 23 probe sets with apparent STAT1-Modifier effects. These probe sets mapped to 21 unique modifiers, 20 of which were kinases, and 1 of which was a phosphatase. Most of these modifiers are supported by literature evidence for some role in STAT1 regulation (Table S6). All predicted modifiers have high priors based on STRING [22], but not all modifiers with high STRING-based priors were predicted by our method. Therefore our method specifically predicts modifiers likely to affect STAT1 transcriptional activity in B cells and related cancers, given some prior knowledge of general protein-protein interaction with STAT1.

To further assess the functional implications of the modifier list predicted by MONSTER

as compared to other interacting partners predicted by STRING, we compared our list of 21 MONSTER-predicted modifiers to the remaining 17 modifiers (kinases and phosphatases only) predicted by STRING but not our model. These lists are too short for a robust statistical analysis, but we observed obvious trends in the functional enrichment of our list as compared to the STRING-only list. For each list, we extracted all KEGG [28] and GO [7] annotations for each modifier using the `GO.db`, `KEGG.db` and `org.Hs.eg.db` packages in Bioconductor [4]. For each term X associated with at least one MONSTER-predicted modifier, we computed the odds ratio of enrichment against the STRING-only list of modifiers, as follows:

$$\text{Odds Ratio}\,(X) = \frac{\%\text{ of MONSTER-predicted modifiers w/ annotation X}}{\%\text{ of STRING-only modifiers w/ annotation X}} \tag{5.1}$$

We attempted to assign p-values to these odds ratios using Fisher's Exact Test, but even strong odds ratio did not appear to be significant due to the aforementioned small sample size, and the general sparsity of pathway annotations. We instead chose to select the annotations with the strongest overall enrichment (Table S8). For GO Biological Process (BP) annotations at least 3 links away from the ontology root, we selected annotations associated with at least 5 of the 21 MONSTER-predicted modifiers, and odds ratio > 2. For KEGG Pathway annotations, which are sparser than GO BP annotations, we selected terms with at least 3 of 21 MONSTER-predicted modifiers, and once again an odds ratio > 2. It is striking that many of the annotations in Table S8 recapitulate known pathways for STAT1, such as "apoptosis", and related terms also enriched among the predicted target genes (Table 3, main manuscript). The list also includes annotations relevant to B cells and related cancers, such as "hemopoiesis" and "hemopoietic or lymphoid organ development".

We also analyzed the available expression data using the MINDY algorithm [24]. We first ranked the target gene probe sets i by global $\text{MI}(i, \text{STAT1})$ estimated by MINDY, and selected the top 1,803 probe sets (the same number of target genes predicted by the MONSTER primary network). We then computed, for each kinase k: $\max|\Delta\text{MI}(i, \text{STAT1})|$ among the top 1,803 target gene probe sets i. We then selected the top 23 modifiers (again, chosen to match the number predicted by MONSTER) based on this summary statistic, and tabulated the results along with supporting literature in Table S7.

Additionally, we analyzed the known functional associations between STAT1 target genes and modifiers predicted by this network model. We extracted from the STRING database [22] functional association scores for all pairs (i, k) in our human B cell input data ($1 \le i \le N,\ 1 \le k \le K$). We recomputed the association scores by excluding the "`expression correlation`" channel to avoid any commonality with our input data that might bias the results. Note that these association scores were *not* used to compute the STAT1-Modifier priors $\mathbf{a}$, and therefore are independent of the input data used by MONSTER. We separated these association scores into those within this network - for which both i was a predicted target of STAT1 and k was a predicted modifier of STAT1 - and those not in this network (all remaining associations). We first analyzed the density of associations in each

category, based on the number of associations contained in STRING vs. the number of possible pairs (i, k). We found that the density of defined associations within our network was approximately 4.5-fold greater within our network (0.0219 vs. 0.0049), and this difference is highly significant according to a Fisher exact test (p-value $<$ 2.2E-16). We also compared the distribution of the defined STRING scores, independent of density, and found that in general the association scores defined for (i, k) pairs within this network were higher than those outside the network (Figure S2, Mann-Whitney p-value $<$ 2.2E-16). Therefore, we conclude that this network predicts STAT1 target genes and modifiers which are functionally coherent.

5.2 Alternate STAT1-Mediated Network Model II

The average network model corresponding to cluster II in Figure S9 linked STAT1 to the negative regulation of 2,031 target gene probe sets corresponding to 1,735 unique genes. This set of genes is heavily enriched for annotations related to metabolic and biosynthentic processes, as shown in Table S10. However, this set of genes contains none of the direct STAT1 targets compiled by Robertson et al. [21]. This model also suggests a repressive role for STAT1 ($\beta_1 = -0.29$), although STAT1 is primarily characterized as an activator in the literature [25, 19]. Therefore we argue that this network model is probably dominated by downstream effects. However, it does suggest that in general STAT1 activity is negatively correlated with cell growth and proliferation in the analyzed expression data. Negative regulation of these functions is a known downstream effect of STAT1 activity in many cell and tumor types [29]. As with all of our network models, we cannot completely rule out indirect or parallel effects, both in terms of the target genes and STAT1 modifiers.

5.3 Alternate STAT1-Mediated Network Model III

We hypothesize that the PWM-biased network model is most likely artifactual or representative of broad cellular trends, rather than STAT1-specific regulation. The value of the β_1 parameter in this model suggests that STAT1 has little direct influence on the predicted target genes, although this could simply mean that STAT1 is completely dependent on upstream modifiers for the regulation of target genes in this network model. The value of β_1 is also within the range of values observed when the input expression data is randomly shuffled, making it difficult to assess the significance of this model. Furthermore, the list of predicted target genes is heavily biased towards high-entropy genes which are unlikely to carry significant biological regulatory signal in our expression set (Figure S9D). This target gene list is also enriched primarily for functional annotations related to the nervous system rather than the immune system, although these functional annotations are still highly significant and would not be predicted using the PWM scores alone (Table S11).

We offer several alternate hypotheses for this model. One possibility is that the network reflects indirect regulatory connections between the modifiers and target genes, but mediated by TFs other than STAT1. This may occur in our procedure because other mediating TFs were left out due to lack of sufficient data. Another possibility is that this model does

represent a STAT1-mediated network, albeit one that is not primarily active in B cells. This model is biased towards the PWM-based priors, and therefore is more likely to identify STAT1 binding sites across all cell types and signaling pathways. A role for STAT1 in regulating gene expression in neurons has been experimentally demonstrated [30, 31]. In this case, the network model parameters may be indicative that STAT1 is being decoupled from the regulation of these target genes by the upstream modifiers in the expression compendium we modeled. Ultimately, we cannot differentiate among these possibilities using existing data.

Supporting Figures

Figure S1: **Accuracy of MONSTER using simulated data.** Accuracy is measured separately for posterior probabilities of TF-Gene connectivity (solid blue lines) and TF-Modifier connectivity (solid green lines), by the area under the ROC curve (AUC) for each network model. The accuracy of network priors (dotted blue and green lines) and TF-Gene expression correlation (dashed blue lines) are shown for comparison. **A.** ROC curves for TF-Gene edges with no expression noise. **B.** ROC curves for TF-Modifier edges with no expression noise. **C.** Accuracy (AUC) for increasing model-wide variance of gene expression residual errors. **D.** Accuracy (AUC) for increasing percentage of uninformative genes as compared to the number of informative genes. **E.** Accuracy (AUC) for increasing percentage of uninformative expression samples as compared to the number of informative samples.

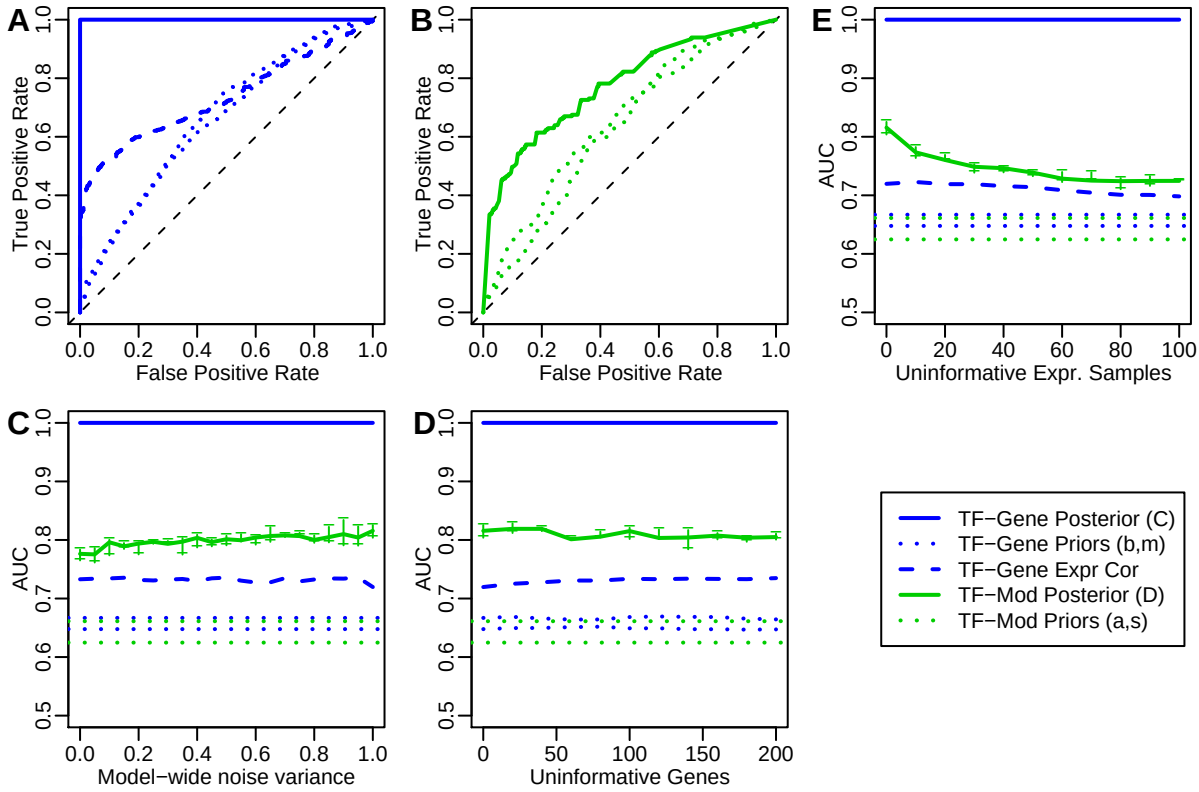


Figure S2: **Modifier-Gene Functional Associations in Primary Model.** STRING functional associations between predicted STAT1 modifiers and target genes within primary network model, compared to functional associations outside the primary network.

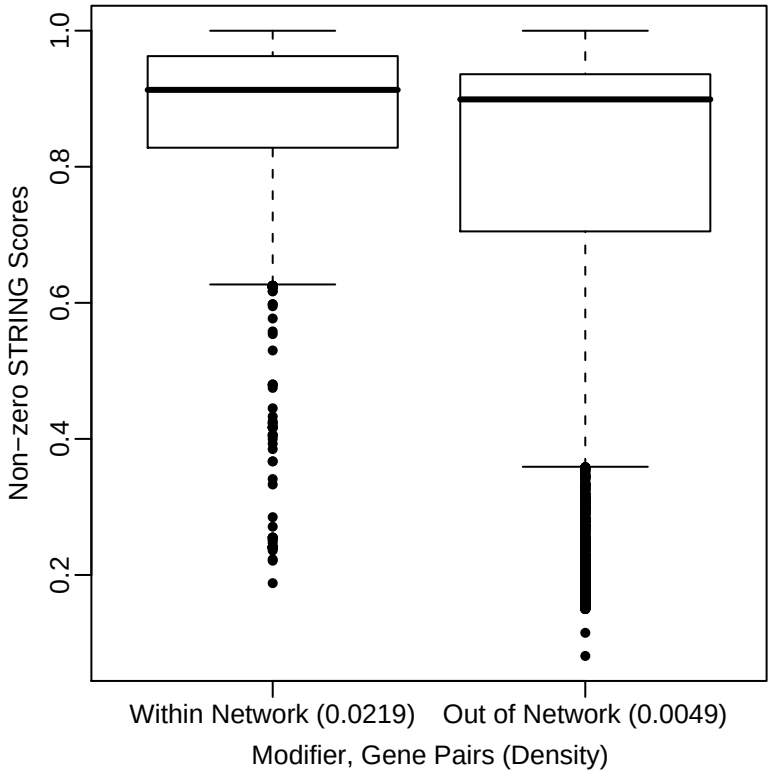


Figure S3: **Yeast Model Priors.** Histograms of network priors used for the yeast application, corresponding to model matrices **m** (left) and **s** (right).

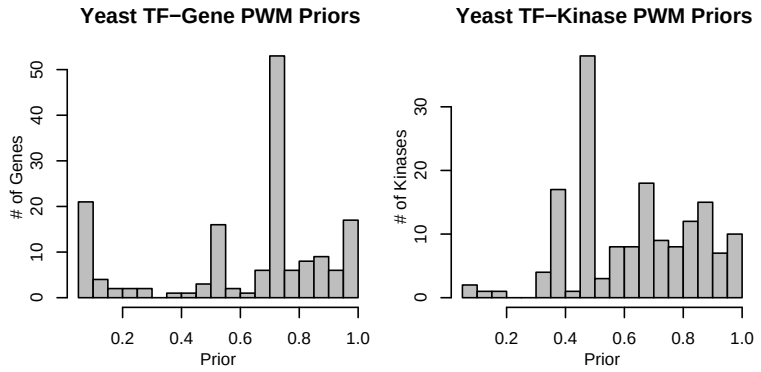


Figure S4: **STAT1 Model Priors.** Histograms of network priors used for the STAT1 application, corresponding to model matrices **b** (left), **m** (center), and **s** (right).

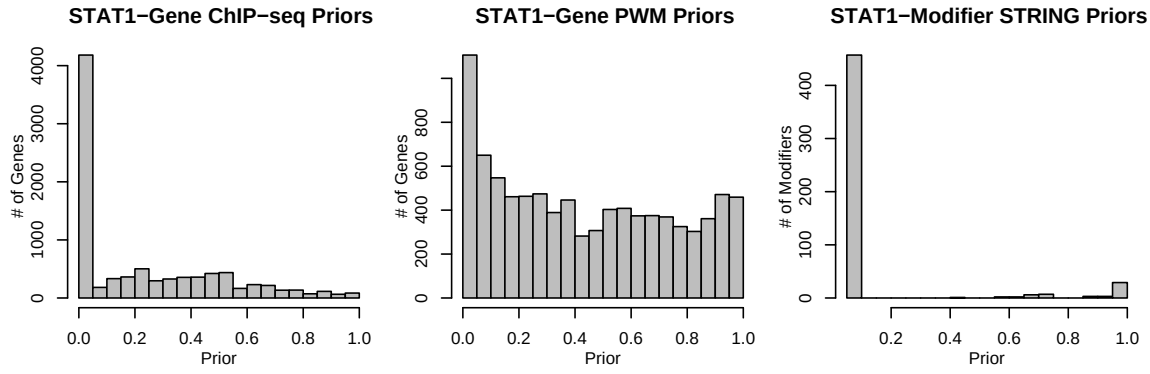


Figure S5: **Mapping STAT1 ChIP-seq Peak Scores to Probabilities.** Mapping was based on output from GLITR algorithm [17], with $X = 0$ corresponding to the highest confidence peaks.

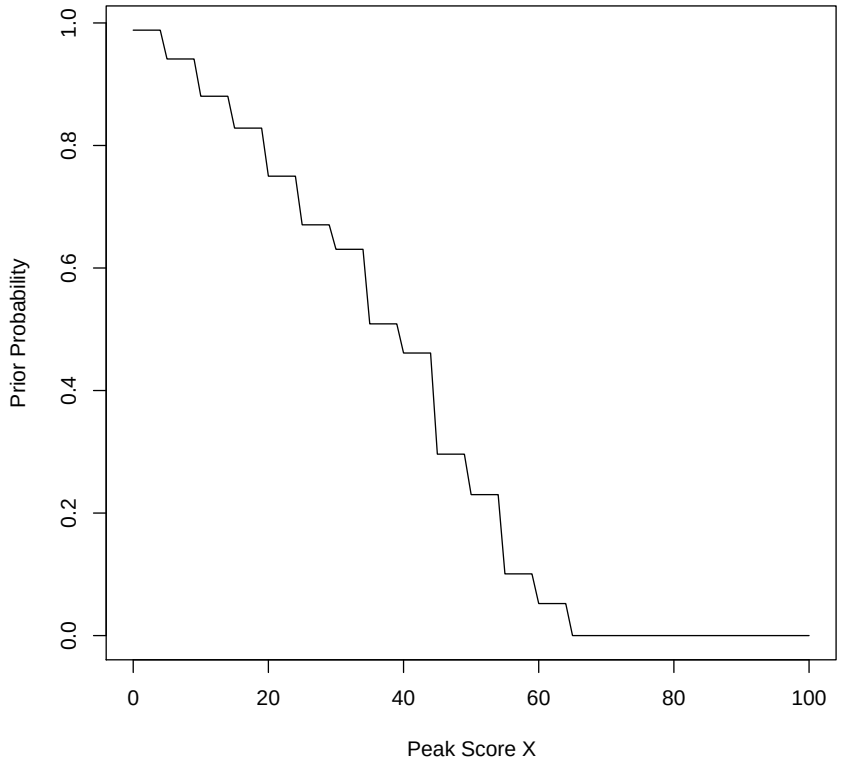


Figure S6: **PWMs for STAT1-Gene priors.** The upper PWM corresponds to the IFN-Stimulated Response Element (ISRE) and the lower PWM corresponds to the IFN-γ-Activated Site (GAS). Both PWMs are represented using WebLogo [32]

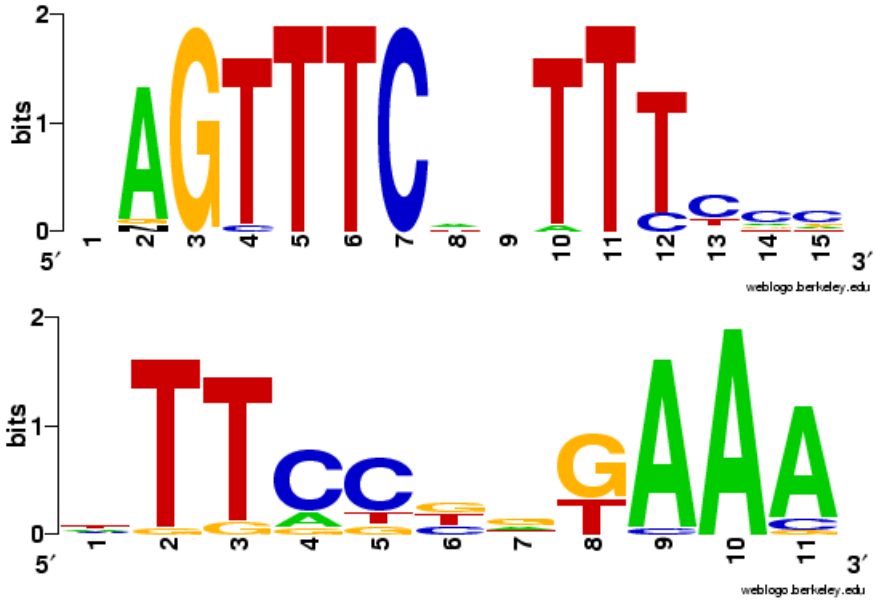


Figure S7: **Hierarchical Clustering of Yeast Models.** Individual network estimates from 100 runs of the model-fitting algorithm were analyzed by hierarchical clustering for: **(A)** the full model and **(B)** the TF-only model. In both model sets, there are no well-separated clusters to indicate distinct solution modes.

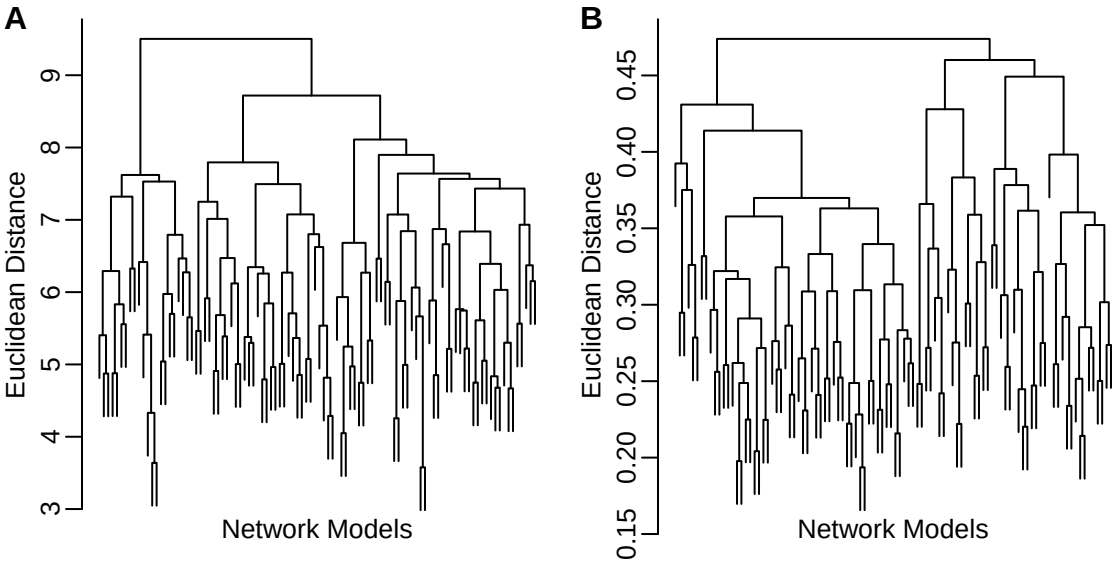


Figure S8: **5-Fold Cross-Validation of Yeast Models.** MSRE values were computed for the full model (green) and the TF-only model (blue) from each of 5 partitions of the expression conditions into 80% training and 20% testing conditions. Mean MSRE on both the test and training data is shown for each cross-validation procedure, with error bars indicated the standard deviation.

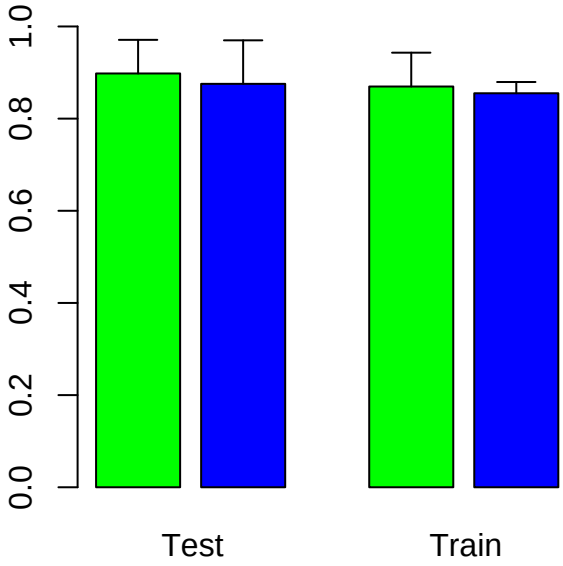


Figure S9: **Hierarchical Clustering of STAT1 Models.** Individual network estimates from 23 runs of the model-fitting algorithm were analyzed by hierarchical clustering. Distinct solution modes are outlined in green, purple, and blue, and numbered I–III, respectively.

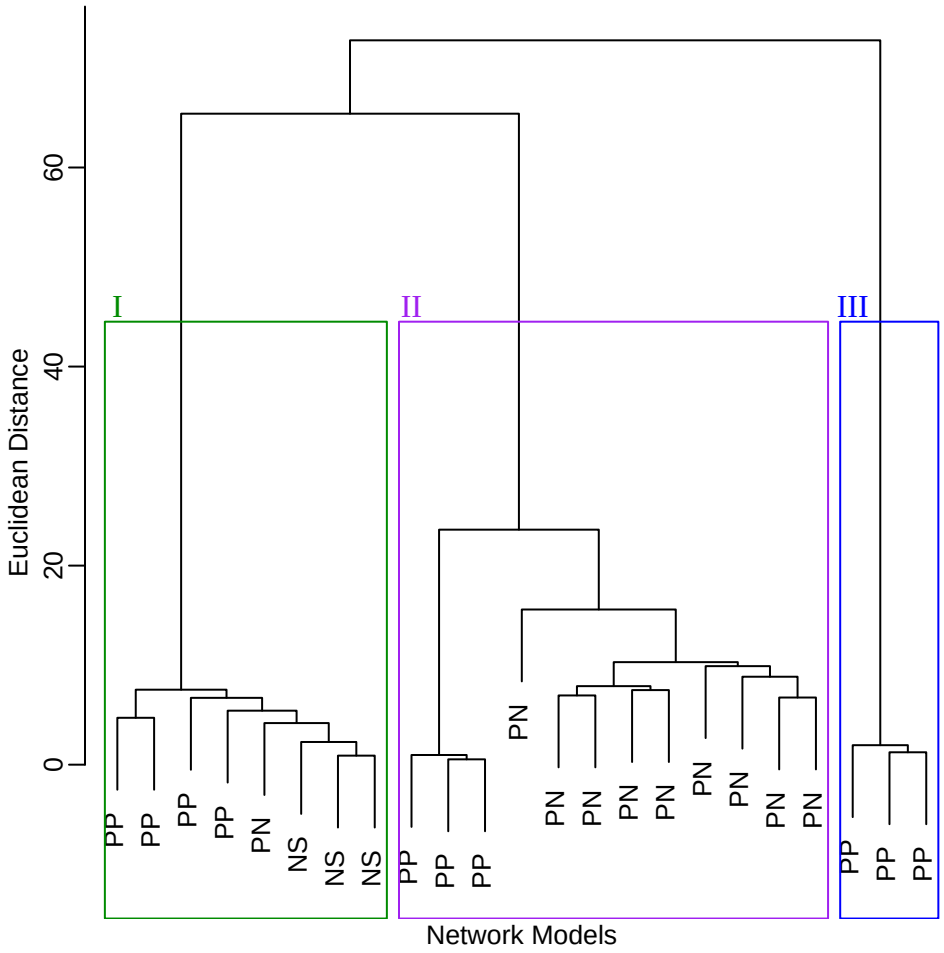
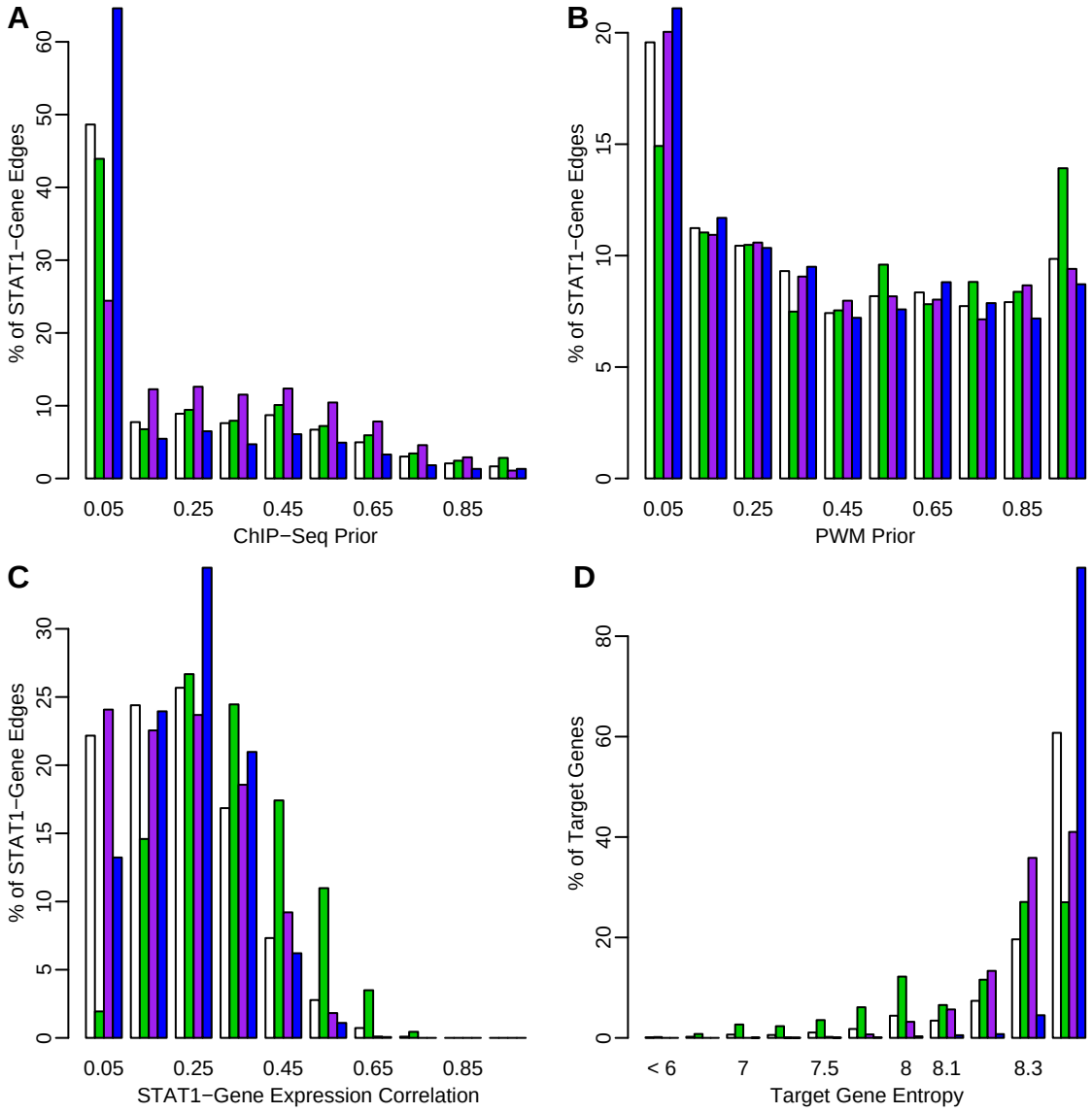


Figure S10: **Distributions of STAT1 Network Properties.** Distributions are shown for: **(A)** ChIP-seq-based priors, **(B)** PWM-based priors, **(C)** STAT1-Target Gene expression correlation, **(D)** Entropy of target genes. White bars represent background distribution for all target genes used as input. Green, purple, and blue bars represent STAT1 target genes predicted by network solution modes I–III, respectively.



Supporting Tables

Table S1: **Input Data Symbols.** Summary of symbols representing components of input data for model-fitting method.

Symbol	Description
N	Number of target genes in input data, also refers to the normal distribution in some sampling equations
J	Number of TFs in input data
K	Number of modifiers in input data
T	Number of expression samples in input data
i	Index over target genes in input data, range $(1, N)$
j	Index over TFs in input data, range $(1, J)$
k	Index over modifiers in input data, range $(1, K)$
t	Index over samples in input data, range $(1, T)$
$\mathbf{g}$	Matrix of target gene expression values for all genes i and all samples t, size $N \times T$
$\mathbf{f}$	Matrix of TF expression values for all TFs j and all samples t, size $J \times T$
$\mathbf{h}$	Matrix of modifier expression values for all modifiers k and samples t, size $K \times T$
$\mathbf{b}$	TF-Gene prior matrix based on experimental binding data for all genes i and TFs j, size $N \times J$
$\mathbf{m}$	TF-Gene prior matrix based on promoter motif analysis for all genes i and TFs j, size $N \times J$
$\mathbf{a}$	TF-Modifier prior matrix based on protein-protein interaction data for all modifiers k and TFs j, size $J \times K$
$\mathbf{s}$	TF-Modifier prior matrix based on substrate profile analysis for all modifiers k and TFs j, size $J \times K$
Ξ	The set of all input data matrices, $(\mathbf{g}, \mathbf{f}, \mathbf{h}, \mathbf{b}, \mathbf{m}, \mathbf{a}, \mathbf{s})$

Table S2: **Model Parameter Symbols.** Summary of symbols representing free parameters in network model.

Symbol	Description
$\boldsymbol{\alpha}$	Vector of basal expression values for target genes, length N
$\boldsymbol{\beta}$	Vector of TF influence parameters, describing linear effect of TF on all target genes, length J
$\boldsymbol{\gamma}$	Matrix of TF-Modifier synergy parameters, describing the sign and magnitude of TF-Modifier synergy terms, size $J \times K$
$\mathbf{C}$	Matrix of indicator variables describing TF-Gene network connectivity, size $N \times J$
$\mathbf{D}$	Matrix of indicator variables describing TF-Modifier network connectivity, size $J \times K$
Φ	General synergy function mapping expression values (f_{jt}, h_{kt}) to synergistic effect of interaction between TF j and modifier k
ε_{it}	Residual error for observed expression of gene i in sample t
σ^2	Model-wide variance of residual errors
$\mathbf{w}$	Vector of weight variables to apply to $\mathbf{b}$ and $\mathbf{m}$ for each TF, length J, values in range $(0, 1)$
$\mathbf{u}$	Vector of weight variables to apply to $\mathbf{a}$ and $\mathbf{s}$ for each modifier, length K, values in range $(0, 1)$
τ_α	Hyperparameter describing the standard deviation of normal prior for all members of $\boldsymbol{\alpha}$
τ_β	Hyperparameter describing the standard deviation of normal prior for all members of $\boldsymbol{\beta}$
τ_γ	Hyperparameter describing the standard deviation of normal prior for all members of $\boldsymbol{\gamma}$
Θ	The set of all non-edge model parameters, $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \sigma^2, \mathbf{w}, \mathbf{u})$

Table S3: **Yeast application kinase & target genes.** All target genes and kinases used as input for yeast application, listed by official gene symbol.

Known MSN2/4 Target Genes						
APJ1	CPR1	CTT1	CYC7	DDR48	ECM4	GAC1
GLK1	GPH1	GRE2	GRX2	GSY2	HSP104	HSP12
HSP26	HSP42	HSP78	HXK1	LAP4	MDJ1	NTH1
PIL1	PNC1	PRX1	PTP2	RAS2	SPS100	SSA4
TPS1	TPS2	TPS3	TRX2	TSA2	TSL1	YDL124W
YGL036W	YML131W	YMR090W	YMR315W	YNL134C		

Decoy Target Genes						
AAT2	ASK1	CAR1	CCT4	CTF3	CUE1	EDC2
EMI5	FMP52	HXT5	ILV1	IML2	KAR4	LAG1
LEA1	LYS9	MEF1	MPP10	NAT1	NIP100	PIB1
PTM1	PUS7	RCR2	RIM4	RIX7	RPN1	RPS14B
RPS29B	RPS4A	RPS6B	RPT3	SMY2	THI20	TRM10
VPS41	VPS62	VRG4	YLF2	YLH47		

Kinase Genes						
AKL1	ARK1	ATG1	BCK1	CDC15	CDC28	CDC5
CKA1	CLA4	CMK1	CMK2	CTK1	DBF2	DUN1
ELM1	FMP48	FRK1	FUS3	GIN4	HAL5	HRK1
HRR25	HSL1	IKS1	IME2	IPL1	IRE1	KCC4
KIN1	KIN2	KIN28	KIN3	KIN4	KIN82	KNS1
KSP1	KSS1	MCK1	MEK1	MKK1	MPS1	NNK1
PBS2	PHO85	PKH2	PKH3	PKP2	PRK1	PRR1
PSK1	PTK2	RAD53	RCK1	RCK2	RIM11	RIM15
SAK1	SAT4	SKM1	SKS1	SKY1	SLT2	SNF1
SPS1	SSK22	SSN3	STE11	STE20	SWE1	TDA1
TOS3	TPK1	TPK2	TPK3	VHS1	YAK1	YCK1
YCK2	YCK3	YGK3	YPL150W			

Table S4: **Significance of Yeast Application AUC Values.** AUC values were computed using TF-Gene edge scores from MONSTER full model $\mathbf{C}^{(+)}$, MONSTER TF-only model $\mathbf{C}^{(-)}$, PWM-based priors $\mathbf{m}$, absolute expression correlation $|\text{cor}(\mathbf{g}, \mathbf{f})|$, and MINDY mutual information analysis. Corresponding ROC curves are drawn in Figure 2 (main manuscript). Permutation p-values estimate the probability that the AUC value arose by chance from a score with no true discriminatory power. Bootstrap p-values estimate the probability that each AUC value is less than the AUC for $\mathbf{C}^{(+)}$ by chance. All p-values are based on 100,000 permutation/bootstrap samples.

			Bootstrap: $< \mathbf{C}^{(+)}$	
Discriminant	**AUC**	**Permutation**	**P-value**	**95% Conf. Int.**
$\mathbf{C}^{(+)}$	0.87	$< 10^{-5}$	-	-
$\mathbf{C}^{(-)}$	0.82	$< 10^{-5}$	0.0018	$(0.015, 0.095)$
$\mathbf{m}$	0.73	$< 10^{-5}$	0.00458	$(0.035, 0.25)$
$\lvert\text{cor}(\mathbf{g}, \mathbf{f})\rvert$	0.63	0.00229	$< 10^{-5}$	$(0.16, 0.33)$
MINDY	0.6	0.01672	$< 10^{-5}$	$(0.18, 0.37)$

Table S5: **Known STAT1 Target Genes.** Gene symbols for verified direct STAT1 targets compiled by Robertson et al. [21], with matching probe sets in the human B cell expression compendium. Genes FCGR1A, FCGR1B, and FCGR1C are all individual known STAT1 targets, but the available expression data contained only a single probe set corresponding to all three of these genes (due to sequence homology), and therefore we count this as a single known gene. $\surd$ indicates STAT1 target is correctly predicted by MONSTER primary model.

Gene Symbol	Probe Set	
ISG15	1107_s_at	$\surd$
	38432_at	
IFI6	1358_s_at	$\surd$
	33832_at	
	34105_f_at	
	37864_s_at	$\surd$
GBP1	35735_at	$\surd$
FCGR1A/B/C	37220_at	$\surd$
CXCL10	431_at	$\surd$
IL6ST	35842_at	$\surd$
	37621_at	$\surd$
IRF1	669_s_at	$\surd$
INDO	36804_at	$\surd$
IFIT2	908_at	$\surd$
	909_g_at	$\surd$
IFIT1	32814_at	$\surd$
	915_at	$\surd$
IFITM1	675_at	$\surd$
IFITM3	41745_at	$\surd$

Gene Symbol	Probe Set	
OAS1	38388_at	$\surd$
	38389_at	$\surd$
FOS	1915_s_at	$\surd$
	1916_s_at	$\surd$
	2094_s_at	
SERPINA3	33825_at	
ISG20	33304_at	
CIITA	35616_at	
	41511_at	$\surd$
MVP	38064_at	$\surd$
NOS2A	1418_at	
	1419_g_at	
	1948_f_at	
CCL2	34375_at	$\surd$
	874_at	
	875_g_at	$\surd$
ICAM1	32640_at	
CD40	35149_at	
	35150_at	
MX1	37014_at	$\surd$

Table S6: **Modifiers Predicted by Primary Network Model.** Listed by gene symbol, with γ_{jk} value and summary of known role in STAT1 regulation. γ_{jk} values marked * indicate a probe set annotated for the same modifier, which corresponds to an incomplete transcript.

Modifier	γ Value	Summary of Supporting Literature
JAK1	0.29	Activator of STAT1 in both IFN-γ and IFN-α/β signaling [33, 19].
LCK	0.16, -0.05*	STAT1 activation during HSV infection of T-Cells is LCK-dependent [34].
RIPK1	-0.15	Competes with STAT1 for binding at TNF-α receptor [35], no known effect on STAT1 TF activity.
SYK	-0.15	STAT1 activation in response to IFN-α is SYK-dependent [36].
BMX	0.12	BMX activates STAT1 to promote target genes, including apoptotic genes [37, 38, 39].
RPS6KA5	-0.10	Promotes STAT1 S727 phosphorylation in response to UVA *in vivo* and phosphorylates this residue directly *in vitro* [40].
CSNK2B	-0.09	*No direct evidence for influence on STAT1.*
JAK3	-0.09	Mediates JAK/STAT signaling in response to IL-2-family cytokines [41]. Specific activation of STAT1 demonstrated in macrophages [42].
TYK2	0.08	Activator of STAT1 in IFN-α/β signaling [33, 19].
MAPK14	0.08	Required for STAT1 S727 phosphorylation in IFN signaling [43]. Phosphorylates STAT1 S727 directly in response to UV stress [44].
AKT1	0.08	Mediates STAT1 S727 phosphorylation in IFN-γ signaling [45].
EIF2AK2	-0.08	Inhibits STAT1 transcriptional activity [46].
CAMK2G	0.07	Phosphorylates STAT1 S727, promotes TF activity [47, 48, 49].
PRKCD	-0.07	Phosphorylates STAT1 S727, promotes TF activity [50, 51, 52, 53].
KIT	0.07	Phosphorylates STAT1 *in vitro*, activates and physically interacts with STAT1 *in vivo* [54, 55].
DUSP3	-0.06, 0.12*	Viral homolog VH1 inhibits STAT1 activity [56], interaction between endogenous DUSP3 and STAT1 has not been studied.
ERBB2	0.06	Interacts with STAT1 *in vitro* [57]. Overexpression in bladder cancer blocks IFN-γ-induced STAT1 activation [58].
FYN	0.06	Required for STAT1 phosphorylation in Angiotensin II-treated vascular smooth muscle cells [59]. Required for STAT1 phosphorylation and DNA binding in EGF-treated JB6 cells [60].
PDGFRA	0.06	Activates STAT1 in response to platelet-derived growth factor [61].
FGFR3	-0.05	Can either activate [62] or inhibit [63] STAT1 TF activity, depending on context. Known to be mutated with differential effect on STAT1 activity in some multiple myeloma cell lines [64].
JAK2	0.05	Activator of STAT1 in IFN-γ signaling [33, 19].

Table S7: **STAT1 Modifiers Predicted by MINDY.** Listed by gene symbol, with γ_{jk} value and summary of known role in STAT1 regulation.

Modifier	max|ΔMI|	Summary of Supporting Literature
MAP4K1	0.4727	*No direct evidence for influence on STAT1.*
MAP3K10	0.4597	*No direct evidence for influence on STAT1.*
CSF2RB	0.4584	Responds to specific cytokines and activates STAT1 in conjunction with JAK2 [65].
TYRO3	0.4573	Indirectly activates STAT1 via the IFN-α receptor in response to GAS6 ligand [66].
FYN	0.4567	Required for STAT1 phosphorylation in Angiotensin II-treated vascular smooth muscle cells [59]. Required for STAT1 phosphorylation and DNA binding in EGF-treated JB6 cells [60].
PPP2CB	0.4518	PP2A-dependence was shown for STAT1 nuclear localization and STAT1 target gene expresssion [67]. PP2A also regulates PRMT1, which in turn regulates STAT1 via methylation [68].
PRKY	0.4469	*No direct evidence for influence on STAT1.*
PPM1F	0.4383	*No direct evidence for influence on STAT1.*
SNF1LK	0.4377	*No direct evidence for influence on STAT1.*
DUSP11	0.4368	*No direct evidence for influence on STAT1.*
PRKCQ	0.4343	*No direct evidence for influence on STAT1.*
MAP3K1	0.4289	*No direct evidence for influence on STAT1.*
PPP2CA	0.4273	PP2A-dependence was shown for STAT1 nuclear localization and STAT1 target gene expresssion [67]. PP2A also regulates PRMT1, which in turn regulates STAT1 via methylation [68].
PTPN9	0.4208	*No direct evidence for influence on STAT1.*
CRKRS	0.417	*No direct evidence for influence on STAT1.*
EPHB6	0.4149	*No direct evidence for influence on STAT1.*
STK16	0.414	*No direct evidence for influence on STAT1.*
PTPN11	0.4136	Dephosphorylates STAT1 to attenuate IFN-γ signaling [69, 70].
CAMK2G	0.4125	Phosphorylates STAT1 S727, promotes TF activity [47, 48, 49].
MAP2K2	0.4117	*No direct evidence for influence on STAT1.*
MTMR4	0.4109	*No direct evidence for influence on STAT1.*
ACVRL1	0.4094	*No direct evidence for influence on STAT1.*
PPP1CC	0.4094	*No direct evidence for influence on STAT1.*

Table S8: **Modifier Pathway Annotations.** Annotations with strong odds ratios when comparing MONSTER-predicted modifiers to other modifiers predicted by STRING.

GO BP Term	**MONSTER Count**	**Odds Ratio**
intracellular signaling cascade	15	5.69
protein kinase cascade	11	3.45
hemopoiesis	5	4.82
apoptosis	5	4.82
cell death	5	4.82
programmed cell death	5	4.82
hemopoietic or lymphoid organ development	5	4.82
protein amino acid phosphorylation	19	2.84
phosphorylation	19	2.84
organ development	10	2.14
system development	10	2.14
KEGG PATHWAY Term		
T cell receptor signaling pathway	3	inf
Tight junction	3	inf
Adipocytokine signaling pathway	5	4.82
Jak-STAT signaling pathway	5	2.29

Table S9: **Properties of STAT1 Network Models.** Model numbers correspond to the clusters identified in Figure S9.

Model	β_1 **(s.d.)**	$\mathbf{w_1}$ **(s.d.)**	**target probe sets**	**target genes**
I	0.24 (0.003)	0.51 (0.003)	1,803	1,559
II	−0.29 (0.07)	0.62 (0.008)	2,031	1,735
III	−0.004 (0.0007)	0.26 (0.0006)	3,190	2,679
(Input)	–	–	*8,973*	*6,975*

Table S10: Significantly enriched annotations (1st column) for gene targets in STAT1 network model II, using all gene targets in the input data as the background, with Bonferroni-corrected p-values (2nd column). As a control, corrected p-values are also shown for the most correlated genes (3rd column), and genes with the highest ChIP-seq and PWM-based priors (4th and 5th columns). '–' indicates a p-value > 1 after Bonferroni correction.

GO Biological Process	**MONSTER**	**Expr**	**ChIP**	**PWM**
nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	5E−38	–	7E−08	–
biopolymer metabolic process	2E−27	–	4E−09	–
DNA metabolic process	4E−24	–	0.045	–
ribonucleoprotein complex biogenesis and assembly	4E−22	–	–	–
mRNA metabolic process	2E−21	–	–	–
mRNA processing	9E−21	–	–	–
RNA splicing	1E−18	–	–	–
DNA replication	3E−18	–	–	–
response to DNA damage stimulus	9E−17	–	0.003	–
DNA repair	4E−16	–	*0.15*	–
ribosome biogenesis and assembly	4E−16	–	–	–
cell cycle phase	1E−15	–	–	–
RNA metabolic process	1E−14	–	0.001	–
M phase	5E−14	–	–	–
oxidative phosphorylation	5E−14	–	–	–
mitotic cell cycle	2E−12	–	–	–
rRNA processing	3E−11	–	–	–
rRNA metabolic process	3E−11	–	–	–
electron transport	3E−11	–	–	–
M phase of mitotic cell cycle	1E−10	–	–	–
ATP synthesis coupled electron transport	4E−10	–	–	–
organelle ATP synthesis coupled electron transport	4E−10	–	–	–
macromolecular complex assembly	6E−10	–	–	–
mitosis	7E−10	–	–	–
DNA-dependent DNA replication	9E−10	–	–	–
cellular component assembly	1E−08	–	–	–
tRNA metabolic process	1E−08	–	–	–
mitochondrial electron transport, NADH to ubiquinone	2E−08	–	–	–
cellular biosynthetic process	5E−08	–	–	–
nuclear mRNA splicing, via spliceosome	7E−08	–	–	–
RNA splicing, via transesterification reactions	7E−08	–	–	–

Continued on next page

Table S10: Enriched annotations for gene targets in STAT1 network model II (continued).

GO Biological Process	MONSTER	Expr	ChIP	PWM
RNA splicing, via transesterification reactions with bulged adenosine as nucleophile	7E−08	–	–	–
macromolecule catabolic process	2E−07	–	–	–
protein-RNA complex assembly	2E−07	–	–	–
organelle organization and biogenesis	8E−07	–	–	–
biopolymer catabolic process	3E−06	–	–	–
cellular macromolecule catabolic process	4E−06	–	–	–
purine nucleotide biosynthetic process	6E−06	–	–	–
ribonucleotide biosynthetic process	1E−05	–	–	–
protein metabolic process	5E−05	–	*0.051*	–
ribonucleotide metabolic process	6E−05	–	–	–
intracellular transport	6E−05	–	–	–
purine nucleotide metabolic process	8E−05	–	–	–
purine ribonucleotide biosynthetic process	1E−04	–	–	–
intracellular protein transport across a membrane	2E−04	–	–	–
protein folding	3E−04	–	–	–
purine ribonucleotide metabolic process	3E−04	–	–	–
cell cycle checkpoint	3E−04	–	–	–
macromolecule biosynthetic process	4E−04	–	–	–
spindle organization and biogenesis	5E−04	–	–	–
cellular catabolic process	5E−04	–	–	–
cellular protein metabolic process	9E−04	–	*0.658*	–
ubiquitin-dependent protein catabolic process	9E−04	–	–	–
tRNA aminoacylation for protein translation	0.001	–	–	–
amino acid activation	0.001	–	–	–
tRNA aminoacylation	0.001	–	–	–
tRNA processing	0.001	–	–	–
modification-dependent macromolecule catabolic process	0.001	–	–	–
microtubule-based process	0.001	–	–	–
modification-dependent protein catabolic process	0.001	–	–	–
proteolysis involved in cellular protein catabolic process	0.001	–	–	–
cellular protein catabolic process	0.002	–	–	–
cellular macromolecule metabolic process	0.002	–	–	–
nucleobase, nucleoside and nucleotide metabolic process	0.003	–	–	–

Continued on next page

Table S10: Enriched annotations for gene targets in STAT1 network model II (continued).

GO Biological Process	MONSTER	Expr	ChIP	PWM
RNA transport	0.004	–	–	–
nucleic acid transport	0.004	–	–	–
nucleobase, nucleoside, nucleotide and nucleic acid transport	0.004	–	–	–
DNA recombination	0.006	–	–	–
double-strand break repair	0.006	–	–	–
RNA localization	0.007	–	–	–
protein catabolic process	0.007	–	–	–
interphase of mitotic cell cycle	0.01	–	–	–
transcription initiation	0.011	–	–	–
nucleoside monophosphate biosynthetic process	0.012	–	–	–
nucleoside monophosphate metabolic process	0.012	–	–	–
interphase	0.012	–	–	–
amino acid metabolic process	0.017	–	–	–
nucleotide biosynthetic process	0.017	–	–	–
sterol biosynthetic process	0.022	–	–	–
mRNA transport	0.023	–	–	–
translational initiation	0.025	–	–	–
recombinational repair	0.026	–	–	–
double-strand break repair via homologous recombination	0.026	–	–	–
nucleoside phosphate metabolic process	0.032	–	–	–
nucleotide metabolic process	0.032	–	–	–
DNA replication initiation	0.038	–	–	–
ribonucleoside monophosphate biosynthetic process	0.039	–	–	–
ribonucleoside monophosphate metabolic process	0.039	–	–	–
coenzyme metabolic process	0.044	–	–	–
regulation of cellular metabolic process	–	–	0.002	–
regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	–	–	0.002	–
ubiquitin cycle	–	–	0.024	–
regulation of RNA metabolic process	–	–	0.025	–
regulation of transcription	–	–	0.03	–
DNA damage response, signal transduction resulting in induction of apoptosis	–	–	0.038	–
regulation of transcription, DNA-dependent	–	–	0.044	–

Continued on next page

Table S10: Enriched annotations for gene targets in STAT1 network model II (continued).

GO Biological Process	**MONSTER**	**Expr**	**ChIP**	**PWM**
negative regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	–	–	0.049	–
inflammatory response	–	–	–	0.045
KEGG Pathway				
Oxidative phosphorylation	2E−19	–	–	–
Proteasome	7E−11	–	–	–
Cell cycle	2E−09	–	–	–
Pyrimidine metabolism	3E−09	–	–	–
Aminoacyl-tRNA biosynthesis	1E−05	–	*0.452*	–
Purine metabolism	3E−05	–	–	–
RNA polymerase	3E−04	–	–	–
DNA polymerase	4E−04	–	–	–
Basal transcription factors	0.003	–	–	–
One carbon pool by folate	0.017	–	–	–
Ubiquitin mediated proteolysis	0.02	–	–	–
Biosynthesis of steroids	0.023	–	–	–

Table S11: Significantly enriched annotations (1st column) for gene targets in STAT1 network model III, using all gene targets in the input data as the background, with Bonferroni-corrected p-values (2nd column). As a control, corrected p-values are also shown for the most correlated genes (3rd column), and genes with the highest ChIP-seq and PWM-based priors (4th and 5th columns). '–' indicates a p-value > 1 after Bonferroni correction.

GO Biological Process	MONSTER	Expr	ChIP	PWM
neurological system process	9E−26	–	–	–
G-protein coupled receptor protein signaling pathway	2E−19	–	–	–
synaptic transmission	2E−17	–	–	–
cell-cell signaling	5E−17	–	–	–
system development	9E−16	–	–	–
transmission of nerve impulse	4E−15	–	–	–
cell surface receptor linked signal transduction	5E−12	0.005	–	–
nervous system development	4E−11	–	–	–
ion transport	3E−10	–	–	–
sensory perception	3E−09	–	–	–
organ development	1E−07	–	–	–
metal ion transport	1E−07	–	–	–
neuron development	8E−07	–	–	–
tissue development	3E−06	–	–	–
neurogenesis	4E−06	–	–	–
axonogenesis	7E−06	–	–	–
sensory perception of light stimulus	8E−06	–	–	–
visual perception	8E−06	–	–	–
generation of neurons	9E−06	–	–	–
G-protein signaling, coupled to cyclic nucleotide second messenger	1E−05	–	–	–
neurite development	2E−05	–	–	–
neuron morphogenesis during differentiation	2E−05	–	–	–
neurite morphogenesis	2E−05	–	–	–
cellular morphogenesis during differentiation	4E−05	–	–	–
cyclic-nucleotide-mediated signaling	6E−05	–	–	–
potassium ion transport	7E−05	–	–	–
neuron differentiation	1E−04	–	–	–
cation transport	2E−04	–	–	–
feeding behavior	2E−04	–	–	–
muscle system process	5E−04	–	–	–
muscle contraction	5E−04	–	–	–
monovalent inorganic cation transport	5E−04	–	–	–
signal transduction	7E−04	–	–	–
G-protein signaling, coupled to cAMP nucleotide second messenger	0.003	–	–	–
second-messenger-mediated signaling	0.003	–	–	–

Continued on next page

Table S11: Enriched annotations for gene targets in STAT1 network model III (continued).

GO Biological Process	**MONSTER**	**Expr**	**ChIP**	**PWM**
cAMP-mediated signaling	0.013	–	–	–
neuropeptide signaling pathway	0.014	–	–	–
sodium ion transport	0.016	–	–	–
skeletal development	0.021	*0.860*	–	–
ectoderm development	0.026	–	–	–
central nervous system development	0.028	–	–	–
brain development	0.041	–	–	–
biopolymer metabolic process	–	–	6E−18	–
nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	–	–	7E−15	–
RNA metabolic process	–	–	1E−08	–
response to DNA damage stimulus	–	–	9E−06	–
ubiquitin cycle	–	–	2E−04	–
regulation of cellular metabolic process	–	–	3E−04	–
DNA metabolic process	–	–	4E−04	–
protein metabolic process	–	–	4E−04	–
RNA biosynthetic process	–	–	0.002	–
DNA repair	–	–	0.002	–
transcription, DNA-dependent	–	–	0.002	–
regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	–	–	0.003	–
cellular protein metabolic process	–	–	0.005	–
regulation of RNA metabolic process	–	–	0.006	–
regulation of transcription	–	–	0.008	–
regulation of transcription, DNA-dependent	–	–	0.009	–
cellular macromolecule metabolic process	–	–	0.009	–
biopolymer modification	–	–	0.019	–
RNA splicing	–	–	0.029	–
mRNA processing	–	–	0.038	–
mRNA metabolic process	–	–	0.044	–
response to wounding	–	–	–	0.048
KEGG Pathway				
Neuroactive ligand-receptor interaction	2E−26	*0.77*	–	–
Calcium signaling pathway	0.001	–	–	–
Cell Communication	0.013	–	–	–
C21-Steroid hormone metabolism	0.042	–	–	–
p53 signaling pathway	–	–	0.045	–

Supporting References

[1] Chen G, Jensen ST, Stoeckert CJ: **Clustering of genes into regulons using integrated modeling-COGRIM**. *Genome Biology* 2007, **8**:R4.

[2] Geman S, Geman D: **Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images**. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1984, **PAMI-6**(6):721–741.

[3] Hartigan JA: *Clustering Algorithms*. New York: John Wiley & Sons, Inc. 1975.

[4] Gentleman RC, Carey VJ, Bates DM, Bolstad B, Dettling M, Dudoit S, Ellis B, Gautier L, Ge Y, Gentry J, Hornik K, Hothorn T, Huber W, Iacus S, Irizarry R, Leisch F, Li C, Maechler M, Rossini AJ, Sawitzki G, Smith C, Smyth G, Tierney L, Yang JYH, Zhang J: **Bioconductor: open software development for computational biology and bioinformatics**. *Genome Biology* 2004, **5**(10):R80.

[5] Levy S, Hannenhalli S: **Identification of transcription factor binding sites in the human genome sequence**. *Mammalian Genome* 2002, **13**(9):510–514.

[6] Harbison CT, Gordon DB, Lee TI, Rinaldi NJ, Macisaac KD, Danford TW, Hannett NM, Tagne J, Reynolds DB, Yoo J, Jennings EG, Zeitlinger J, Pokholok DK, Kellis M, Rolfe PA, Takusagawa KT, Lander ES, Gifford DK, Fraenkel E, Young RA: **Transcriptional regulatory code of a eukaryotic genome**. *Nature* 2004, **431**(7004):99–104.

[7] Hong EL, Balakrishnan R, Dong Q, Christie KR, Park J, Binkley G, Costanzo MC, Dwight SS, Engel SR, Fisk DG, Hirschman JE, Hitz BC, Krieger CJ, Livstone MS, Miyasato SR, Nash RS, Oughtred R, Skrzypek MS, Weng S, Wong ED, Zhu KK, Dolinski K, Botstein D, Cherry JM: **Gene Ontology annotations at SGD: new data sources and annotation methods**. *Nucleic Acids Research* 2008, **36**(S1):D577–581.

[8] Saunders NFW, Kobe B: **The Predikin webserver: improved prediction of protein kinase peptide specificity using structural information**. *Nucleic Acids Research* 2008, **36**(Web Server issue):W286–290.

[9] Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Hannett NM, Harbison CT, Thompson CM, Simon I, Zeitlinger J, Jennings EG, Murray HL, Gordon DB, Ren B, Wyrick JJ, Tagne J, Volkert TL, Fraenkel E, Gifford DK, Young RA: **Transcriptional regulatory networks in Saccharomyces cerevisiae**. *Science* 2002, **298**(5594):799–804.

[10] Ptacek J, Devgan G, Michaud G, Zhu H, Zhu X, Fasolo J, Guo H, Jona G, Breitkreutz A, Sopko R, McCartney RR, Schmidt MC, Rachidi N, Lee S, Mah AS, Meng L, Stark

MJR, Stern DF, Virgilio CD, Tyers M, Andrews B, Gerstein M, Schweitzer B, Predki PF, Snyder M: **Global analysis of protein phosphorylation in yeast**. *Nature* 2005, **438**(7068):679–684.

[11] von Mering C, Jensen LJ, Snel B, Hooper SD, Krupp M, Foglierini M, Jouffre N, Huynen MA, Bork P: **STRING: known and predicted protein-protein associations, integrated and transferred across organisms**. *Nucleic acids research* 2005, **33**(Database issue):D433–437.

[12] Barrett T, Troup DB, Wilhite SE, Ledoux P, Rudnev D, Evangelista C, Kim IF, Soboleva A, Tomashevsky M, Marshall KA, Phillippy KH, Sherman PM, Muertter RN, Edgar R: **NCBI GEO: archive for high-throughput functional genomic data**. *Nucleic Acids Research* 2009, **37**(Database issue):D885–890.

[13] Basso K, Margolin AA, Stolovitzky G, Klein U, Dalla-Favera R, Califano A: **Reverse engineering of regulatory networks in human B cells**. *Nature Genetics* 2005, **37**(4):382–390.

[14] Irizarry RA, Bolstad BM, Collin F, Cope LM, Hobbs B, Speed TP: **Summaries of Affymetrix GeneChip probe level data**. *Nucleic Acids Research* 2003, **31**(4):e15.

[15] Maglott D, Ostell J, Pruitt KD, Tatusova T: **Entrez Gene: gene-centered information at NCBI**. *Nucleic Acids Research* 2007, **35**(Database issue):D26–31.

[16] Rozowsky J, Euskirchen G, Auerbach RK, Zhang ZD, Gibson T, Bjornson R, Carriero N, Snyder M, Gerstein MB: **PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls**. *Nature Biotechnology* 2009, **27**:66–75.

[17] Tuteja G, White P, Schug J, Kaestner KH: **Extracting transcription factor targets from ChIP-Seq data**. *Nucleic Acids Research* 2009, **37**(17):e113.

[18] Pruitt KD, Tatusova T, Maglott DR: **NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins**. *Nucleic Acids Research* 2007, **35**(Database issue):D61–65.

[19] Platanias LC: **Mechanisms of type-I- and type-II-interferon-mediated signalling**. *Nature Reviews Immunology* 2005, **5**(5):375–386.

[20] Wingender E, Dietze P, Karas H, Knppel R: **TRANSFAC: a database on transcription factors and their DNA binding sites**. *Nucleic Acids Research* 1996, **24**:238–241.

[21] Robertson G, Hirst M, Bainbridge M, Bilenky M, Zhao Y, Zeng T, Euskirchen G, Bernier B, Varhol R, Delaney A, Thiessen N, Griffith OL, He A, Marra M, Snyder M, Jones S: **Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing**. *Nature Methods* 2007, **4**(8):651–657.

[22] Jensen LJ, Kuhn M, Stark M, Chaffron S, Creevey C, Muller J, Doerks T, Julien P, Roth A, Simonovic M, Bork P, von Mering C: **STRING 8–a global view on proteins and their functional interactions in 630 organisms**. *Nucleic Acids Research* 2009, **37**(Database issue):D412–416.

[23] Ewens WJ, Grant GR: *Statistical methods in bioinformatics*. Springer 2005.

[24] Wang K, Saito M, Bisikirska BC, Alvarez MJ, Lim WK, Rajbhandari P, Shen Q, Nemenman I, Basso K, Margolin AA, Klein U, Dalla-Favera R, Califano A: **Genome-wide identification of post-translational modulators of transcription factor activity in human B cells**. *Nature Biotechnology* 2009, **27**(9):829–839.

[25] Darnell JE: **STATs and gene regulation**. *Science* 1997, **277**(5332):1630–1635.

[26] Shuai K, Liu B: **Regulation of JAK-STAT signalling in the immune system**. *Nature Reviews Immunology* 2003, **3**(11):900–911.

[27] Schug J, Schuller W, Kappen C, Salbaum JM, Bucan M, Stoeckert CJ: **Promoter features related to tissue specificity as measured by Shannon entropy**. *Genome Biology* 2005, **6**(4):R33.

[28] Ogata H, Goto S, Sato K, Fujibuchi W, Bono H, Kanehisa M: **KEGG: Kyoto Encyclopedia of Genes and Genomes**. *Nucleic Acids Research* 1999, **27**:29–34.

[29] Regis G, Pensa S, Boselli D, Novelli F, Poli V: **Ups and downs: the STAT1:STAT3 seesaw of Interferon and gp130 receptor signalling**. *Seminars in Cell & Developmental Biology* 2008, **19**(4):351–359.

[30] Cho HJ, Kim S, Jin SM, Hwang E, Kim YS, Huh K, Mook-Jung I: **IFN-gamma-induced BACE1 expression is mediated by activation of JAK2 and ERK1/2 signaling pathways and direct binding of STAT1 to BACE1 promoter in astrocytes**. *Glia* 2007, **55**(3):253–262.

[31] Cho HJ, Jin SM, Son SM, Kim YW, Hwang JY, Hong HS, Mook-Jung I: **Constitutive JAK2/STAT1 activation regulates endogenous BACE1 expression in neurons**. *Biochemical and Biophysical Research Communications* 2009, **386**:175–180.

[32] Crooks GE, Hon G, Chandonia J, Brenner SE: **WebLogo: a sequence logo generator**. *Genome Research* 2004, **14**(6):1188–1190.

[33] Brierley MM, Fish EN: **Stats: multifaceted regulators of transcription**. *Journal of Interferon & Cytokine Research* 2005, **25**(12):733–744.

[34] Lund TC, Garcia R, Medveczky MM, Jove R, Medveczky PG: **Activation of STAT transcription factors by herpesvirus Saimiri Tip-484 requires p56lck**. *Journal of Virology* 1997, **71**(9):6677–6682.

[35] Wang Y, Wu TR, Cai S, Welte T, Chin YE: **Stat1 as a component of tumor necrosis factor alpha receptor 1-TRADD signaling complex to inhibit NF-kappaB activation**. *Molecular and Cellular Biology* 2000, **20**(13):4505–4512.

[36] Tassiulas I, Hu X, Ho H, Kashyap Y, Paik P, Hu Y, Lowell CA, Ivashkiv LB: **Amplification of IFN-alpha-induced STAT1 activation and inflammatory function by Syk and ITAM-containing adaptors**. *Nature Immunology* 2004, **5**(11):1181–1189.

[37] Saharinen P, Ekman N, Sarvas K, Parker P, Alitalo K, Silvennoinen O: **The Bmx tyrosine kinase induces activation of the Stat signaling pathway, which is specifically inhibited by protein kinase Cdelta**. *Blood* 1997, **90**(11):4341–4353.

[38] Wen X, Lin HH, Shih HM, Kung HJ, Ann DK: **Kinase activation of the non-receptor tyrosine kinase Etk/BMX alone is sufficient to transactivate STAT-mediated gene expression in salivary and lung epithelial cells**. *The Journal of Biological Chemistry* 1999, **274**(53):38204–38210.

[39] Chen K, Huang L, Kung H, Ann DK, Shih H: **The role of tyrosine kinase Etk/Bmx in EGF-induced apoptosis of MDA-MB-468 breast cancer cells**. *Oncogene* 2004, **23**(10):1854–1862.

[40] Zhang Y, Cho Y, Petersen BL, Zhu F, Dong Z: **Evidence of STAT1 phosphorylation modulated by MAPKs, MEK1 and MSK1**. *Carcinogenesis* 2004, **25**(7):1165–1175.

[41] Habib T, Nelson A, Kaushansky K: **IL-21: a novel IL-2-family lymphokine that modulates B, T, and natural killer cell responses**. *The Journal of Allergy and Clinical Immunology* 2003, **112**(6):1033–1045.

[42] Sareila O, Korhonen R, Krpnniemi O, Nieminen R, Kankaanranta H, Moilanen E: **Janus kinase 3 inhibitor WHI-P154 in macrophages activated by bacterial endotoxin: differential effects on the expression of iNOS, COX-2 and TNF-alpha**. *International Immunopharmacology* 2008, **8**:100–108.

[43] Goh KC, Haque SJ, Williams BR: **p38 MAP kinase is required for STAT1 serine phosphorylation and transcriptional activation induced by interferons**. *The EMBO Journal* 1999, **18**(20):5601–5608.

[44] Kovarik P, Mangold M, Ramsauer K, Heidari H, Steinborn R, Zotter A, Levy DE, Mller M, Decker T: **Specificity of signaling by STAT1 depends on SH2 and C-terminal domains that regulate Ser727 phosphorylation, differentially affecting specific target gene expression**. *The EMBO Journal* 2001, **20**(1-2):91–100.

[45] Nguyen H, Ramana CV, Bayes J, Stark GR: **Roles of phosphatidylinositol 3-kinase in interferon-gamma-dependent phosphorylation of STAT1 on serine 727 and activation of gene expression**. *The Journal of Biological Chemistry* 2001, **276**(36):33361–33368.

[46] Wong AH, Tam NW, Yang YL, Cuddihy AR, Li S, Kirchhoff S, Hauser H, Decker T, Koromilas AE: **Physical association between STAT1 and the interferon-inducible protein kinase PKR and implications for interferon and double-stranded RNA signaling pathways**. *The EMBO Journal* 1997, **16**(6):1291–1304.

[47] Nair JS, DaFonseca CJ, Tjernberg A, Sun W, Darnell JE, Chait BT, Zhang JJ: **Requirement of Ca2+ and CaMKII for Stat1 Ser-727 phosphorylation in response to IFN-gamma**. *Proceedings of the National Academy of Sciences of the United States of America* 2002, **99**(9):5971–5976.

[48] Xu W, Nair JS, Malhotra A, Zhang JJ: **B cell antigen receptor signaling enhances IFN-gamma-induced Stat1 target gene expression through calcium mobilization and activation of multiple serine kinase pathways**. *Journal of Interferon & Cytokine Research* 2005, **25**(2):113–124.

[49] Wang L, Tassiulas I, Park-Min K, Reid AC, Gil-Henn H, Schlessinger J, Baron R, Zhang JJ, Ivashkiv LB: **'Tuning' of type I interferon-induced Jak-STAT1 signaling by calcium-dependent kinases in macrophages**. *Nature Immunology* 2008, **9**(2):186–193.

[50] Uddin S, Sassano A, Deb DK, Verma A, Majchrzak B, Rahman A, Malik AB, Fish EN, Platanias LC: **Protein kinase C-delta (PKC-delta) is activated by type I interferons and mediates phosphorylation of Stat1 on serine 727**. *The Journal of Biological Chemistry* 2002, **277**(17):14408–14416.

[51] Deb DK, Sassano A, Lekmine F, Majchrzak B, Verma A, Kambhampati S, Uddin S, Rahman A, Fish EN, Platanias LC: **Activation of protein kinase C delta by IFN-gamma**. *Journal of Immunology* 2003, **171**:267–273.

[52] DeVries TA, Kalkofen RL, Matassa AA, Reyland ME: **Protein kinase Cdelta regulates apoptosis via activation of STAT1**. *The Journal of Biological Chemistry* 2004, **279**(44):45603–45612.

[53] Zhao K, Li D, Zhao Q, Huang Y, Silverman RH, Sims PJ, Chen G: **Interferon-alpha-induced expression of phospholipid scramblase 1 through STAT1 requires the sequential activation of protein kinase Cdelta and JNK**. *The Journal of Biological Chemistry* 2005, **280**(52):42707–42714.

[54] Deberry C, Mou S, Linnekin D: **Stat1 associates with c-kit and is activated in response to stem cell factor**. *The Biochemical Journal* 1997, **327**:73–80.

[55] Corbacioglu S, Kilic M, Westhoff M, Reinhardt D, Fulda S, Debatin K: **Newly identified c-KIT receptor tyrosine kinase ITD in childhood AML induces ligand-independent growth and is responsive to a synergistic effect of imatinib and rapamycin**. *Blood* 2006, **108**(10):3504–3513.

[56] Najarro P, Traktman P, Lewis JA: **Vaccinia virus blocks gamma interferon signal transduction: viral VH1 phosphatase reverses Stat1 activation**. *Journal of Virology* 2001, **75**(7):3185–3196.

[57] Jones RB, Gordus A, Krall JA, MacBeath G: **A quantitative protein interaction network for the ErbB receptors using protein microarrays**. *Nature* 2006, **439**(7073):168–174.

[58] Su W, Tu I, Hu S, Yeh H, Shieh D, Chen T, Su W: **HER-2/neu raises SHP-2, stops IFN-gamma anti-proliferation in bladder cancer**. *Biochemical and Biophysical Research Communications* 2007, **356**:181–186.

[59] Venema RC, Venema VJ, Eaton DC, Marrero MB: **Angiotensin II-induced tyrosine phosphorylation of signal transducers and activators of transcription 1 is regulated by Janus-activated kinase 2 and Fyn kinases and mitogen-activated protein kinase phosphatase 1**. *The Journal of Biological Chemistry* 1998, **273**(46):30795–30800.

[60] He Z, Tang F, Ermakova S, Li M, Zhao Q, Cho Y, Ma W, Choi H, Bode AM, Yang CS, Dong Z: **Fyn is a novel target of (-)-epigallocatechin gallate in the inhibition of JB6 Cl41 cell transformation**. *Molecular Carcinogenesis* 2008, **47**(3):172–183.

[61] Valgeirsdttir S, Paukku K, Silvennoinen O, Heldin CH, Claesson-Welsh L: **Activation of Stat5 by platelet-derived growth factor (PDGF) is dependent on phosphorylation sites in PDGF beta-receptor juxtamembrane and kinase insert domains**. *Oncogene* 1998, **16**(4):505–515.

[62] Sahni M, Ambrosetti DC, Mansukhani A, Gertner R, Levy D, Basilico C: **FGF signaling inhibits chondrocyte proliferation and regulates bone development through the STAT-1 pathway**. *Genes & Development* 1999, **13**(11):1361–1366.

[63] Krejci P, Prochazkova J, Bryja V, Jelinkova P, Pejchalova K, Kozubik A, Thompson LM, Wilcox WR: **Fibroblast growth factor inhibits interferon gamma-STAT1 and interleukin 6-STAT3 signaling in chondrocytes**. *Cellular Signalling* 2009, **21**:151–160.

[64] Ronchetti D, Greco A, Compasso S, Colombo G, Dell'Era P, Otsuki T, Lombardi L, Neri A: **Deregulated FGFR3 mutants in multiple myeloma cell lines with t(4;14): comparative analysis of Y373C, K650E and the novel G384D mutations**. *Oncogene* 2001, **20**(27):3553–3562.

[65] Brizzi MF, Aronica MG, Rosso A, Bagnara GP, Yarden Y, Pegoraro L: **Granulocyte-macrophage colony-stimulating factor stimulates JAK2 signaling pathway and rapidly activates p93fes, STAT1 p91, and STAT3 p92 in polymorphonuclear leukocytes**. *The Journal of Biological Chemistry* 1996, **271**(7):3562–3567.

[66] Rothlin CV, Ghosh S, Zuniga EI, Oldstone MBA, Lemke G: **TAM receptors are pleiotropic inhibitors of the innate immune response**. *Cell* 2007, **131**(6):1124–1136.

[67] Fielhaber JA, Han Y, Tan J, Xing S, Biggs CM, Joung K, Kristof AS: **Inactivation of mammalian target of rapamycin increases STAT1 nuclear content and transcriptional activity in alpha4- and protein phosphatase 2A-dependent fashion**. *The Journal of Biological Chemistry* 2009, **284**(36):24341–24353.

[68] Christen V, Duong F, Bernsmeier C, Sun D, Nassal M, Heim MH: **Inhibition of alpha interferon signaling by hepatitis B virus**. *Journal of Virology* 2007, **81**:159–165.

[69] Wu TR, Hong YK, Wang X, Ling MY, Dragoi AM, Chung AS, Campbell AG, Han Z, Feng G, Chin YE: **SHP-2 is a dual-specificity phosphatase involved in Stat1 dephosphorylation at both tyrosine and serine residues in nuclei**. *The Journal of Biological Chemistry* 2002, **277**(49):47572–47580.

[70] You M, Yu DH, Feng GS: **Shp-2 tyrosine phosphatase functions as a negative regulator of the interferon-stimulated Jak/STAT pathway**. *Molecular and Cellular Biology* 1999, **19**(3):2416–2424.