Supplementary Information

Hill *et al.*, "Bayesian inference of signaling network topology in a cancer cell line".

1 Proteomics and validation experiments

1.1 Reverse phase protein arrays

Reverse phase protein array (RPPA) assays were carried out as previously described [1, 2]. Breast cancer cell line MDA-MB-468 was cultured in its optimal media to a logarithm growth phase. Time courses were carried out at eight time points (5, 15, 30, 60, 90, 120, 180, 240 minutes) in triplicate, under four growth conditions (0, 5, 10, 20ng/ml EGF). Cellular proteins were denatured by 1% SDS (with beta-mercaptoethanol) and diluted in five 2-fold serial dilutions in dilution buffer (lysis buffer containing 1% SDS). Serial diluted lysates were arrayed on nitrocellulose-coated FAST slides (Whatman, Inc) by Aushon 2470 Arrayer (Aushon BioSystems). Total 5808 array spots were arranged on each slide including the spots corresponding to positive and negative controls prepared from mixed cell lysates or dilution buffer, respectively.

Each slide was probed with a validated primary antibody (Table S1) plus a biotin-conjugated secondary antibody. Only antibodies with a Pearson correlation coefficient between RPPA and western blotting of greater than 0.7 were used in reverse phase protein array study. Antibodies with a single or dominant band on western blotting were further assessed by direct comparison to RPPA using cell lines with differential protein expression or modulated with ligands/inhibitors or siRNA for phospho- or structural proteins, respectively. Extensive validation data for the antibodies used are presented in [2].

The signal obtained was amplified using a DakoCytomation-catalyzed system (Dako) and visualized by DAB colorimetric reaction. The slides were scanned, analyzed, and quantified using a customized-software Microvigene (VigeneTech Inc.) to generate spot intensity.

Each dilution curve was fitted with a logistic model ("Supercurve Fitting" developed by the Department of Bioinfomatics and Computational Biology in MD Anderson Cancer Center, `http://bioinformatics.mdanderson.org/OOMPA`). This fits a single curve using all the samples (i.e., dilution series) on a slide with the signal intensity as the response variable and the dilution steps are independent variable. The fitted curve is plotted with the signal intensities both observed and fitted - on the y-axis and the log2-concentration of proteins on the x-axis for diagnostic purposes. The protein concentrations of each set of slides were then normalized by median polish, which was corrected across samples by the linear expression values using the median expression levels of all antibody experiments to calculate a loading correction factor for each sample. Logged averages over RPPA triplicates (Figure S9) were used for all network analyses.

1.2 Validation experiments

1.2.1 RPPA (Figure 4, Main Text)

The breast cancer cell line MDA-MB-468 was seeded at 90% confluency in 96-well plates at a density of 10,000 cells per well with 8% FBS-RPMI medium and allowed to attach. Cells were depleted of serum for 12 hours prior to treatment with the MEK inhibitor GSK1120212 (GlaxoSmithKline Inc.) at 0uM, 0.625uM, 2.5uM and 10uM or AKT inhibitor GSK690693B (GlaxoSmithKline Inc.) at 0uM, 0.625uM, 2.5uM and 10uM for 4 hours in each case. Cells were stimulated (EGF 20 ng/mL) prior to lysis in RPPA lysis buffer. Phosphoprotein profiling was carried out 0,5,15,30,60,90,120,180 minutes after EGF stimulus using RPPA as described above.

2 Methods

Below we give further details of the model marginal likelihood, our exact approach for inference of posterior edge probabilities, the 'network prior', empirical Bayes setting of prior strength parameter λ and cross-validation. All computations were carried out in MATLAB R2009a using software that is available at `http://mukherjeelab.nki.nl/DBN`.

2.1 Marginal Likelihood

Let p denote the number of components under study (signaling proteins in the present work) and T denote number of time points sampled. DBNs associate a random variable with each of the p components at each time point. Let these pT variables be denoted by X_i^t and $X^t = \left(X_1^t, \ldots, X_p^t\right)$ be the corresponding random vector at time t. Thus, the full, "unrolled" graph, with each X_i^t explicitly represented as a vertex (Figure S1b) contains pT vertices, at T time points. First-order Markov and stationarity assumptions mean the unrolled DBN can be described by a smaller graph with two vertices for each component under study, representing adjacent time points (the "collapsed DBN"; see Figure S1c). This latter graph structure, which we denote by G in what follows, is the object of inference.

The DBN graph G permits factorisation of the "global" joint probability distribution over all variables (the likelihood) into a product of "local" conditional distributions, with each variable depending only on its parents at the previous time point. This sparse, time-invariant dependence leads to a model with far fewer parameters than a model with a full, time-varying dependence structure. The factorisation is shown in Equation 1 in Main Text and below.

$$p(\mathbf{X} \mid G, \Theta) = \prod_{i=1}^{p} p(X_i^1 \mid \psi_i) \prod_{t=2}^{T} p(X_i^t \mid X_{\pi_G(i)}^{t-1}, \theta_i) \tag{1}$$

where $\mathbf{X} = \left(X^1, \ldots, X^T\right)$ is all data, $\pi_G(i) \subseteq \{1, \ldots, p\}$ is an index set for parents of protein i

according to G, $X^t_{\pi_G(i)} = \left\{ X^t_j \mid j \in \pi_G(i) \right\}$ is data for the parents of protein i at time t, $\theta_i \subseteq \Theta$ are parameters for the conditional distribution of X^t_i and $\psi_i \subseteq \Theta$ are parameters for X^1_i.

Since the marginal $p(X^1_i)$ does not depend on graph G, it is omitted in what follows. In the interests of notational simplicity, we introduce the vector $X^+_i = \left(X^2_i, \ldots, X^T_i\right)^\mathsf{T}$ to denote all data for protein i in the "current" (second) time slice of the "collapsed DBN" (Figure S1c) and $X^-_i = \left(X^1_i, \ldots, X^{T-1}_i\right)^\mathsf{T}$ to denote corresponding data in the "previous" (first) time slice. This allows us to remove the product over time above and express the likelihood in the following simple form :

$$p(\mathbf{X} \mid G, \Theta) \propto \prod_{i=1}^{p} p(X^+_i \mid X^-_{\pi_G(i)}, \theta_i) \tag{2}$$

(up to a multiplicative constant that does not depend on graph G)

The conditionals $p(X^t_i \mid X^{t-1}_{\pi_G(i)}, \theta_i)$ constituting the likelihood are taken to be Gaussian. These describe the dependence of child nodes on their parents and can be thought of as regression models, with parents and child corresponding to covariates and response respectively. The $2^{|\pi_G(i)|} - 1$ regression coefficients, forming vector β_i, and variance σ^2_i, constitute parameters θ_i. Considering all n pairs of adjacent time points (i.e. the "collapsed DBN"), we let $\mathbf{B}_i$ be a $n \times (2^{|\pi_G(i)|} - 1)$ local design matrix with columns corresponding to parents and products of parents as described in Main Text. Then we have

$$p(X^+_i \mid X^-_{\pi_G(i)}, \theta_i) = \mathrm{Normal}(\mathbf{B}_i \beta_i, \sigma^2_i I_n) \tag{3}$$

where I_n is the $n \times n$ identity matrix. We note that X^+_i and each column of the design matrix $\mathbf{B}_i$ are standardized to have zero mean and unit variance. Following [3, 4] we use the reference prior $p(\sigma^2_i) \propto \sigma_i^{-2}$ for local variances and a $\mathrm{Normal}(\mathbf{0}, n\sigma^2_i(\mathbf{B}_i{}^\mathsf{T}\mathbf{B}_i)^{-1})$ prior for local coefficients β_i. The latter has variance proportional to the variance of the least squares estimate for β_i. This prior is related to Zellner's g-prior [5] and has attractive invariance properties under rescaling of the data (see [6] for details). We note that this formulation, unlike the widely-used "BGe" score [7], has no free, user-set parameters. Assuming prior parameter independence [7] then yields the following integral for the marginal likelihood,

$$\begin{aligned} p(\mathbf{X} \mid G) &= \int p(\mathbf{X} \mid G, \Theta) p(\Theta \mid G) \, \mathrm{d}\Theta \\ &\propto \prod_{i=1}^{p} \iint p(X^+_i \mid X^-_{\pi_G(i)}, \beta_i, \sigma^2_i) p(\beta_i \mid G, \sigma^2_i) p(\sigma^2_i) \, \mathrm{d}\beta_i \mathrm{d}\sigma^2_i. \end{aligned} \tag{4}$$

The coefficients β_i can be integrated out as a normal integral and σ^2_i as an inverse-gamma integral. This yields a closed-form marginal likelihood as shown in Main Text and reproduced below (multiplicative constants that do not depend on G are omitted) :

$$p(\mathbf{X} \mid G) \propto \prod_{i=1}^{p} (1+n)^{-(2^{|\pi_G(i)|}-1)/2} \left(X^+_i{}^\mathsf{T} X^+_i - \frac{n}{n+1} X^+_i{}^\mathsf{T} \mathbf{B}_i \left(\mathbf{B}_i{}^\mathsf{T} \mathbf{B}_i \right)^{-1} \mathbf{B}_i{}^\mathsf{T} X^+_i \right)^{-\frac{n}{2}}. \tag{5}$$

Evaluating the above marginal likelihood for any given graph structure G requires the inversion of a $\left(2^{|\pi_G(i)|}-1\right) \times \left(2^{|\pi_G(i)|}-1\right)$ matrix ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i$, which may be ill-conditioned or even singular, especially when n is small relative to in-degree $|\pi_G(i)|$. In this work, an in-degree restriction of $|\pi_G(i)| \leq d_{\max}$ with $d_{\max} = 4$ (see below) helps to avoid these numerical issues and precludes extreme ill-conditioning. In general we advise the selection of a $d_{\max}$ satisying $2^{|\pi_G(i)|} - 1 \leq n$ (or $|\pi_G(i)| \leq n$ for a linear model without interaction terms). In addition to this restriction, and if necessary, regularization can be performed to improve conditioning. Where necessary (see below) we regularized through addition of a ridge term to the matrix ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i$. That is, ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i$ is replaced by ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i + \alpha\mathbf{I}$ in Equation 5, where $\mathbf{I}$ is the identity matrix and $\alpha > 0$. This gives in effect a reduction in the prior variance of regression parameters $\boldsymbol{\beta}$. Regularization is performed if the condition number of ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i$ is above a threshold τ (we set $\tau = 10^4$). The tuning parameter α is set in an automated, iterative fashion by incrementing its value (starting at zero) until the condition number of ${\mathbf{B}_i}^{\mathsf{T}}\mathbf{B}_i + \alpha\mathbf{I}$ is below threshold τ. We note that scale invariance of the g-prior formulation is lost if regularization is used.

For the analyses presented in the Main Text, regularization was only invoked for the combination of interaction terms with larger parent set sizes ($|\pi_G(i)|$=3 or 4). No regularization was needed for any of the models without interaction terms or for $|\pi_G(i)| < 3$. When invoked, the magnitude of reduction in prior variance was small; for the breast cancer cell line study we observed a 5% decrease on average for $|\pi_G(i)|$=4 and 2% decrease on average for $|\pi_G(i)|$=3. Whilst the percentage decrease is (unsurprisingly) larger for $|\pi_G(i)|$=4, these parent sets have less influence on the posterior than parents sets of size $|\pi_G(i)|$=3 because they are more heavily penalized by the marginal likelihood (due to their larger model complexity). Re-running the analysis on the breast cancer cell line data without performing any regularization gave good agreement with the results reported in Figure 3a (Main Text) with all subsequently validated edges retained.

We note that other prior formulations, for example a standard ridge prior or the "BGe" prior, ameliorate or do not suffer from matrix conditioning problems and could be good choices in very small n settings, or when it is desirable to relax the in-degree restriction. In this context, the "BGe" prior has the attractive property that it does not require a matrix inversion, but it requires the setting of more hyperparameters.

2.2 Exact inference

We are interested in calculating posterior probabilities of edges $e = (a, b)$ in the graph G. Note that $a, b \in \{1, \ldots, p\}$, with a, b representing variables from the "previous" and "current" time slices respectively (i.e. the first and second time slices of the "collapsed DBN"). For simplicity, we use $e = (a, b)$ in what follows and leave the time associated with the vertices implicit. The posterior probability of the edge is calculated by averaging over the space of all possible graphs $\mathcal{G}$ [8],

$$P(e \mid \mathbf{X}) = \sum_{G \in \mathcal{G}} \mathbb{1}_{\{e \in G\}} P(G \mid \mathbf{X}). \tag{6}$$

where $P(G \mid \mathbf{X})$ is the posterior distribution over graphs.

However, for the DBNs used here, it is possible to utilize a variable selection approach to efficiently calculate posterior edge probabilities exactly, thereby increasing confidence in results whilst avoiding the need for expensive convergence diagnostics. Specifically, for each "response" variable X_i^+ in the "current" time slice, we calculate posterior scores for subsets $\pi(i) \subseteq \{1, \ldots, p\}$ of potential predictors from the previous time slice ("parent sets"),

$$\begin{aligned} P(\pi(i) \mid X^-, X_i^+) &\propto p(X_i^+ \mid X^-, \pi(i)) P(\pi(i) \mid X^-) \\ &= p(X_i^+ \mid X_{\pi(i)}^-) P(\pi(i)) \end{aligned} \tag{7}$$

where $X^- = \left(X_1^-, \ldots, X_p^-\right)$ denotes all data in the previous time slice and $P(\pi(i))$ is a prior distribution over sets of predictor variables. The likelihood $p(X_i^+ \mid X_i^-, \pi\left(i\right), \theta_i)$ is as in Equation 3 above, with parameter priors for β_i and σ_i^2 also as described above. Integrating out parameters θ_i then results in the marginal likelihood $p(X_i^+ \mid X^-, \pi\left(i\right))$,

$$\begin{aligned} p(X_i^+ \mid X^-, \pi\left(i\right)) \propto (1+n)^{-(2^{|\pi(i)|}-1)/2} \Big(& {X_i^+}^{\mathsf{T}} X_i^+ \\ & - \frac{n}{n+1} {X_i^+}^{\mathsf{T}} \mathbf{B}_i \left({\mathbf{B}_i}^{\mathsf{T}} \mathbf{B}_i \right)^{-1} {\mathbf{B}_i}^{\mathsf{T}} X_i^+ \Big)^{-\frac{n}{2}} . \end{aligned} \tag{8}$$

This is a marginal likelihood of the form that appears in Bayesian variable selection for regression problems (see e.g. [9]). We now discuss how model averaging in the variable selection sense can be used to make inference about DBN structure. We perform model averaging to calculate the posterior probability of a specific predictor variable X_a^- being in the model for response variable X_b^+. In terms of the DBN framework, we can think of this as an edge $e = (a, b)$. Then we have,

$$P(e \mid X^-, X_b^+) = \sum_{\pi(b)} \mathbb{1}_{\{a \in \pi(b)\}} P(\pi(b) \mid X^-, X_b^+) \tag{9}$$

where the summation is over all possible sets of predictor variables for variable X_b^+. If the network prior $P(G)$ factorizes into a product of local priors over parents sets $\pi_G(i)$ for each variable,

$$P(G) = \prod_{i=1}^{p} P(\pi_G(i)) \tag{10}$$

(the network prior used here satisfies this property; see below) then posterior edge probabilities calculated via Equation 6 are identical to those calculated via Equation 9. This is essentially due to the modular form of the marginal likelihood in Equation 5 and the guaranteed acyclicity of the DBNs employed here. Indeed, this equivalence holds for any modular scoring function and modular network prior used with DBNs (with edges only allowed forwards in time), as we now demonstrate.

If the marginal likelihood $P(\mathbf{X} \mid G)$ has a modular form (as in Equation 5) and the network prior $P(G)$ satisfies Equation 10, then the posterior over graphs $P(G \mid \mathbf{X})$ is simply a product of posteriors

over subsets of predictors for each variable (Equation 7), as specified by the edge structure of G:

$$P(G \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid G)P(G)}{\sum_G P(\mathbf{X} \mid G)P(G)} \tag{11}$$

$$= \frac{\prod_{i=1}^{p} p(X_i^+ \mid X_{\pi_G(i)}^-)P(\pi_G(i))}{\sum_{\pi_G(1)} \cdots \sum_{\pi_G(p)} \prod_{i=1}^{p} p(X_i^+ \mid X_{\pi_G(i)}^-)P(\pi_G(i))} \tag{12}$$

$$= \prod_{i=1}^{p} \frac{p(X_i^+ \mid X_{\pi_G(i)}^-)P(\pi_G(i))}{\sum_{\pi_G(i)} p(X_i^+ \mid X_{\pi_G(i)}^-)P(\pi_G(i))} \tag{13}$$

$$= \prod_{i=1}^{p} P(\pi_G(i) \mid X^-, X_i^+). \tag{14}$$

We can now observe that edge probabilities calculated via averaging over the full graph space (Equation 6) equal those calculated from a variable selection approach (Equation 9). In particular, for an edge $e = (a, b)$,

$$\begin{aligned}
P(e \mid \mathbf{X}) &= \sum_G \mathbb{1}_{\{e \in G\}} P(G \mid \mathbf{X}) \\
&= \sum_G \mathbb{1}_{\{e \in G\}} \prod_{i=1}^{p} P(\pi_G(i) \mid X^-, X_i^+) \\
&= \sum_{\pi_G(1)} \cdots \sum_{\pi_G(p)} \mathbb{1}_{\{e \in G\}} \prod_{i=1}^{p} P(\pi_G(i) \mid X^-, X_i^+) \\
&= \left(\sum_{\pi(b)} \mathbb{1}_{\{a \in \pi(b)\}} P(\pi(b) \mid X^-, X_b^+) \right) \prod_{\substack{1<i<p \\ i \neq b}} \left(\sum_{\pi(i)} P(\pi(i) \mid X^-, X_i^+) \right) \\
&= \sum_{\pi(b)} \mathbb{1}_{\{a \in \pi(b)\}} P(\pi(b) \mid X^-, X_b^+) \\
&= P(e \mid X^-, X_b^+)
\end{aligned}$$

Note that, for each variable X_i^+, the space of possible subsets of predictors is of size 2^p (as opposed to 2^{p^2} for the full graph space). Hence it is much more efficient to calculate edge probabilities via Equation 9 than Equation 6. However, the problem is still exponential in p. Motivated by the fact that typically only a small number of key upstream kinases are critical for the activation of any given network component (see e.g. [10]), and following previous work [11, 12], we enforce a maximum in-degree constraint and only consider up to four proteins jointly influencing a target. Whilst, under this restriction, the full graph space size still grows quicker than exponential in p, the space of subsets of predictors becomes polynomial in p. This enables exact calculation of edge probabilities via Equation 9.

We compared the results reported in the Main Text, obtained by exact calculation with maximum in-degree of four, with results obtained by (i) exact calculation with maximum in-degree increased to

five, and (ii) a MCMC-based analysis with no restriction on in-degree. We found very good agreement between the regimes (Figure S2), showing that results were not dependent on the sparsity restriction.

The equivalence between posterior probabilities calculated via averaging over the full graph space and those calculated via a variable selection approach does not merely hold for single edge probabilities, but for any graph feature that can be fully specified at a local parent set level. That is, equivalence holds if an indicator function $I(G)$, specifying whether graph G has a feature of interest or not, can be expressed as a product of local indicator functions over parent sets,

$$I(G) = \prod_{j=1}^{p} I_j(\pi_G(j)). \tag{15}$$

For example, for single edge probailities (Equation 6), we have $I(G) = \mathbb{1}_{\{e=(a,b)\in G\}}$, $I_b(\pi_G(b)) = \mathbb{1}_{\{a\in\pi_G(b)\}}$ and $I_j(\pi_G(j)) = 1$ for $j \neq b$. Features satisfying this local factorisation include existence of sets of edges, non-existence of sets of edges and in-degree related features. However, the equivalence does not extend to arbitrary graph features.

2.3 Network prior

Following [12, 13], we take prior distribution $P(G)$ to be of the form

$$P(G) \propto \exp(\lambda f(G)) \tag{16}$$

where λ is a strength parameter, weighting the contribution of the prior and $f(G)$ is a real-valued function over graphs, scoring the degree to which graphs concord with our prior beliefs. We use signaling maps, from online repositories (`cellsignal.com`, `stke.sciencemag.org`) and the literature [14, 15], to obtain a set of edges we may expect to see in an inferred network. The edge set includes indirect edges via components not included in our study and edges between protein phosphoforms and isoforms. We denote this set of *a priori* expected edges by E^*. The same edge set is used for both cell lines in our study. We let $f(G) = -|E(G)\backslash E^*|$ where $E(G)$ is the set of edges contained in G. That is, $f(G)$ is the number of edges in G that are not included in our expected edge set E^*. Thus, our prior does not actively promote any particular edge, but rather penalizes unusual edges. The edge set E^* is given in Figure S3 and includes links (i) from EGFR to AKTs, MEK, JNK, p70, LKB1, p38, PI3K and STAT3/5; (ii) from MEK to ERK, and from ERK to p70, p90 and TSC2; (iii) from mTOR, PDK1 and PI3K to AKTs, and from AKTs to GSK3, mTOR, p70 and TSC2 [16]; (iv) from PI3K to PDK1 and mTOR; (v) from LKB1 to AMPK [17], and from AMPK to mTOR [18] and TSC2; (v) from mTOR, PDK1 and TSC2 [19] to p70, and from p70 to EGFR; (vi) from cJun to JNK; (vii) from mTOR to STAT3/5 [20] and from TSC2 to mTOR [21]; (viii) from PDK1 to p90 [22], and from p90 and GSK3 to TSC2.

The prior used here satisfies the modular form of Equation 10. The expected edge set E^* can be represented by p index sets $\pi^*(i) \subseteq \{1, \ldots, p\}$ for the parents of each "response" variable X_i^+, as

given by E^*. That is, $\pi^*(i) = \{j \mid (j,i) \in E^*\}$. Then, Equation 10 holds if we define the prior over parent sets to be $P(\pi_G(i)) \propto \exp(\lambda f_i(\pi_G(i)))$ where $f_i(\pi_G(i)) = -|\pi_G(i) \backslash \pi^*(i)|$.

2.4 Empirical Bayes

The prior strength parameter λ is set objectively by an empirical Bayes approach, as discussed in Main Text. We present here some further details. λ is set by maximising the following marginal likelihood,

$$\begin{aligned} p(\mathbf{X} \mid \lambda) &= \mathbb{E}\left[p(\mathbf{X} \mid G)\right]_{P(G \mid \lambda)} \\ &= \sum_G p(\mathbf{X} \mid G) P(G \mid \lambda). \end{aligned} \tag{17}$$

Following similar arguments as in Section 2.2, Equation 17 above can be rewritten in terms of summations over subsets of predictors $\pi(i)$ as follows,

$$p\left(\mathbf{X} \mid \lambda\right) = \prod_i \sum_{\pi(i)} p\left(X_i^+ \mid X_{\pi(i)}^-\right) P\left(\pi\left(i\right) \mid \lambda\right). \tag{18}$$

This allows the marginal likelihood to be calculated efficiently within the exact inference framework used here. The marginal likelihood score is calculated over a grid of hyperparameter values and those resulting in the largest score are used in the analysis.

3 Results

3.1 Simulation study

We carried out a simulation study in which data were generated from known graphs and the results of network inference were compared with the true data-generating graphs. Thus, the data-generating graphs provided a "gold standard" against which to assess the analyses. Mirroring the protein signaling study, we formed DBNs with 20 vertices (corresponding to proteins) in each time slice and simulated 4 time courses of 8 time points each. We carried out inference as described above and in Main Text, and used the network prior shown in Figure S3.

Data-generating graphs were created so as to agree only partially with the prior used. This was done using a random, Erdös-Renyi-like approach. In particular, an edge set $E(G)$ for a data-generating graph G was created from the prior graph edge set E^* using a two-step process. First, edges contained in the prior edge set E^* were removed at random, leaving only 20 of the original 74 edges. Second, 10 edges that were not originally in E^* were added; these were chosen uniformly at random. This process gave randomly generated graphs with 30 edges. For each such randomly generated graph, 10 of the edges were not in the prior, while 54 of the edges in the prior were not in the data-generating graph. This created a scenario in which a non-trivial proportion of the prior graph used did not agree with the data-generating graph.

Data were then generated from a given graph by ancestral sampling (through time), using a Gaussian model with interaction terms (see Equation 3). The zero-order or bias term was always included, and the remaining terms were independently included with probability 0.5 (subject to each parent in the graph being represented by at least one term; this ensured that the data model was faithful to the graph). Model regression coefficients were sampled from a uniform distribution over $[-1,-0.1]\cup[0.1,1]$ and were independent of time. Root nodes (initial time point) were also sampled from this uniform distribution and Gaussian noise was set at a variance of 0.5. This can be thought of as simulating data from a sparse vector autoregressive (VAR) model. For example, if protein i has two parents in the data-generating graph, proteins j_1 and j_2, then for the initial time point $t=1$, we take $X_i^1 \sim \mathrm{Uniform}([-1,-0.1]\cup[0.1,1])$ and for $t\in\{2,\ldots,8\}$ we have

$$X_i^t = \beta_0 + \gamma_1\beta_1 X_{j_1}^{t-1} + \gamma_2\beta_2 X_{j_2}^{t-1} + \gamma_3\beta_3 X_{j_1}^{t-1} X_{j_2}^{t-1} + \epsilon_{it} \tag{19}$$

where $\epsilon_{it} \sim \mathcal{N}(0,0.5)$, regression coefficients $\beta_k \sim \mathrm{Uniform}([-1,-0.1]\cup[0.1,1])$ and γ_k are independent Bernoulli random variables taking value one with probability 0.5, thereby selecting which terms are included. Sampled γ's are rejected if they are not faithful to the graph: as an example, for this three protein illustration, $\gamma_1 = 1$ and $\gamma_2 = \gamma_3 = 0$ would be rejected as it would mean that contrary to the graph protein j_2 is not actually a parent of protein i.

We used network inference for DBNs, as described above, using a model with interaction terms and an informative network prior with empirical Bayes, to calculate posterior edge probabilties $P(e|\mathbf{X})$. Thresholding these probabilities at level τ produces an edge set $E_\tau = \{e \mid P(e \mid \mathbf{X}) \geq \tau\}$, which can be compared to the true data-generating graph edge set to calculate number of true positives (correct edges called at threshold τ) and number of false positives (incorrect edges called at threshold τ). A receiver operating characteristic (ROC) curve is created by plotting number of true positives against number of false positives for varying thresholds. Area under the ROC curve (AUC) provides a measure of network inference accuracy with higher values indicating better performance.

We generated a total of 25 random graphs, as described above. Empirical Bayes setting of prior strength parameter resulted in an average value of $\lambda = 3.54 \pm 0.34$ over the 25 experiments. Figure 2 in Main Text shows average ROC curves and average AUC values obtained are shown in Figure S4. We also show results using DBN inference without interaction terms and/or a flat prior over graph space (i.e. $P(G) = \mathrm{constant}$), a baseline correlational analysis (thresholded absolute correlation coefficients between variables at adjacent time points), variable selection via an ℓ_1-penalized (Lasso) regression approach [23] (implemented using Matlab package `glmnet` [24]), and several previously proposed network inference approaches for time course data. These approaches were: a Gaussian graphical model approach using functional data analysis and shrinkage-based estimation [25] (implemented using R package `GeneNet`), a non-Bayesian approach for inferring DBNs [26] (implemented using R package `G1DBN`), and a non-parametric Bayesian approach using Gaussian processes [27] (implemented using Matlab package `gp4grn`). (All Matlab and R packages were used with default settings). All the above methods resulted in a set of edge scores (e.g. posterior edge probabilities,

regression coefficients, partial correlations etc.) that were thresholded to produce ROC curves. Since Gaussian graphical models are undirected graphs, the inferred networks from this approach were compared with the data-generating graph without edge directions. We note the Lasso approach also provides a single graph (i.e. a point estimate) by selecting all those edges with non-zero regression coefficients.

Mean AUCs ($\pm$ SD) for network inference with an informative prior and flat prior were 0.93 ± 0.03 and 0.84 ± 0.05 respectively. The baseline correlational analysis, Lasso and previously proposed network inference approaches had mean AUC values ranging from 0.61 to 0.80. Hence, we see that the network prior provides significant gains in sensitivity and specificity, even though a non-trivial proportion of information in the prior is, by design of the simulation study, not in agreement with the data-generating model. We also note that, due to its inability to model combinatorial interplay, the proposed DBN method without interaction terms is outperformed by the method including interaction terms.

3.2 Synthetic yeast network study

The objectivity of simulation studies depends on how biologically realistic the data-generating model is. In our simulations above, the data generating model is not biologically realistic and also matches the inference model. While conclusions can be made regarding, for example, the utility of incorporating prior information, performance on this data is not likely to reflect performance on real data, and the comparisons between methods are biased in favour of those that are based on the data-generating model. Due to these limitations, simulation strategies that are more realistic, for example based on non-linear systems of ODEs, can improve objectivity of assessments and comparisons [11].

Recent work by [28] provides another approach for assessing structure learning performance using biologically realistic data and a known gold-standard network. A gene regulatory network was synthetically constructed in the yeast *Saccharomyces cerevisiae*. This network, called the IRMA network, is composed of five genes and six regulatory interactions (plus a protein-protein interaction). These interactions include feedback mechanisms and the network was designed so that the five genes are negligibly affected by genes not in the network. Since the network is known, gene expression (mRNA) data obtained from the system can be used to assess performance of structure learning algorithms; we apply this approach here.

We use data from the "switch-off' experiments [28], which consists of 18 time points (every 10 minutes up to 3 hours) and is averaged over four replicates. As noted by [28], it is less likely that the protein-protein interaction can be recovered from the mRNA level data, so we assess performance based on the network with six edges. In order to investigate the effect of including prior information, we formed prior networks that partially agree with the IRMA network as follows. First, the prior edge set E^* is taken to include the six edges in the IRMA network and all self-loop edges. Second, three edges, chosen at random, are added to E^*. Third, two edges in the IRMA network, chosen at random, are removed from E^*. This results in prior networks containing seven edges (plus self-loop edges),

four of which are in the IRMA network, and the IRMA network contains two edges that are not in the prior.

We applied exact network inference for DBNs as described above and in Main Text, using an informative network prior (generated as just described) with strength parameter set by empirical Bayes and linear model with interaction terms. Empirical Bayes resulted in an average value of $\lambda = 4.72 \pm 3.75$ over 25 different randomly generated prior networks. As in the simulation study, posterior edge probabilities were used to generate ROC curves and AUC scores. Table 1 in Main Text shows AUC scores obtained for the same methods and regimes considered above in the simulation study. AUC scores for DBN inference with an informative prior and flat prior (and with interaction terms) were 0.82 ± 0.04 (mean$\pm$SD) and 0.75 respectively. As in the simulation study above, we observe gains in accuracy through use of a network prior, even though the prior is only in partial agreement with the true network. We also see that inclusion of interaction terms provides an improvement in performance. The Gaussian processes method performs comparably to the DBN approach described in this Chapter, but is significantly more computationally intensive (for the five gene network, 10 iterations of the DBN approach takes approximately 1.5 seconds, compared to 160 seconds for the Gaussian processes method). The baseline correlational analysis, Lasso, Gaussian graphical model and non-Bayesian DBN inference approaches did not perform well, with AUC values ranging from 0.39 to 0.5 (i.e. no improvement over a random classifier).

3.3 Network model for breast cancer cell line MDA-MB-468

3.3.1 Robustness to specification of network prior structure

Here we give a few extra details to those given in Main Text. We investigated robustness to changes in the prior graph (Figure 3c in Main Text and Figure S7). This was done by perturbing the prior graph and comparing inferred posterior edge probabilities to those reported above. Perturbations were made by making a number of edge removals and additions, keeping the total number of edges constant (changing one edge in the prior network consists of an edge removal and then an edge addition; the total number of edge removals and additions made is called the 'structural hamming distance' or SHD). Results are robust to changes in the prior graph: for example, changing one third of the edges (25 out of 74 edges; SHD equal to 50), gave edge probabilities that showed a correlation of 0.88 ± 0.03 with those reported (mean Pearson correlation $\pm$ SD; calculated from 25 perturbed prior graphs).

3.3.2 Robustness to data perturbation

In addition to investigating robustness of results reported to prior specification, we sought also to investigate the robustness to perturbation of the data. We did so by removing parts of the data and replacing with the average of adjacent time points. The data consists of four time series of eight time points each. We removed data, for all 20 proteins simultaneously, from between 1 and 4 time-point/condition combinations: this corresponded to removing between 1/32 and 1/8 of the data. These

deletions represent a non-trivial change to a small data set. Figure S8a shows Pearson correlation coefficients between edge probabilties reported in Main Text (from unperturbed data; Figure 3a) and those inferred from perturbed data. We observe good agreement between the edge probabilities, demonstrating that results are not overly sensitive to changes to the data. For example, perturbing 1/8 of the data resulted in a correlation coefficient of 0.83 ± 0.05 (calculated from 25 perturbed datasets).

We also considered the case of completely removing *all* data at one of the eight time points. This reduces the amount of data by almost 15% and also changes the interval between sampled time points. Removing each of the six intermediate time points in turn, we again compared results obtained to the edge probabilities reported above. We found an average correlation coefficient of 0.81 ± 0.06. This demonstrates that the results reported are robust, even when 1/8 of the data is removed and time intervals are substantially changed. Indeed, even when deleting two complete time points (randomly selected), i.e. 1/4 of the data, we found reasonable agreement with the results reported (correlation coefficient of 0.71 ± 0.08).

For both analyses above, the prior strength parameter was fixed to the value used in the original analysis (λ=3).

3.3.3 Predictive capability and model fit

As an empirical check of model fit and predictive capability, we carried out leave-one-out-cross-validation (LOOCV) using inferred posterior parent set probabilties (Equation 7). We note that in the present setting LOOCV cannot be expected to guide detailed model choice or design. This is due to the fact that for small data sets LOOCV alone is limited in its ability to choose between models or regimes. Rather, our aim in using LOOCV was simply to highlight any egregious mismatch between data and model. In the time-course setting it is also difficult to interpret LOOCV results in terms of absolute error: rather we compared LOOCV error against baseline analyses.

At each iteration, one of the n data samples was removed and the exact inference procedure described above and in Main Text used to learn the posterior distribution over parent sets $P(\pi(i)|X^-, X_i^+)$ from the remaining $n-1$ training samples. We denote the training data for variable i by X^- and X_i^+ and the corresponding held-out sample by Y^- and Y_i^+. Given the training data, we can predict the expected value of Y_i^+ from Y^- by model averaging (see e.g. [9]),

$$\mathbb{E}\left[Y_i^+ \mid Y^-, X^-, X_i^+\right] = \sum_{\pi(i)} \mathbb{E}\left[Y_i^+ \mid Y^-, X^-, X_i^+, \pi(i)\right] P(\pi(i) \mid X^-, X_i^+) \tag{20}$$

with

$$\mathbb{E}\left[Y_i^+ \mid Y^-, X^-, X_i^+, \pi(i)\right] = \frac{n}{n+1} \tilde{\mathbf{B}}_i \left(\mathbf{B}_i{}^{\mathsf{T}} \mathbf{B}_i\right)^{-1} \mathbf{B}_i{}^{\mathsf{T}} X_i^+ \tag{21}$$

where $\mathbf{B}_i$ is the $(n-1) \times (2^{|\pi(i)|}-1)$ design matrix for variable i, including products of parents, and $\tilde{\mathbf{B}}_i$ is the corresponding $1 \times (2^{|\pi(i)|}-1)$ design matrix for the held-out sample. Full network inference, including empirical Bayes learning of the hyperparamter, was carried out at each cross-validation

iteration.

Figure S8b shows these predictions compared with those from (i) the single highest-scoring graph under the posterior distribution over DBN structures (this is the *maximum a posteriori* or MAP counterpart to the Bayesian model averaging approach we propose); (ii) the proposed DBN inference method with a fully linear regression model without interaction terms; (iii) an ℓ_1-penalized regression (Lasso) approach in which parents are inferred via variable selection; (iv) a baseline autoregressive model in which each variable depends only on itself at the previous time point; and (v) a baseline non-sparse linear model in which each protein depends on all parents. For (i), (iii), (iv) and (v) predictions are made using Equation 21 only. The proposed method shows lower LOOCV error relative to the baseline models and to the MAP model, and performs comparably to DBN inference with a fully linear model and to Lasso regression.

References

[1] Tibes R, et al. (2006) Reverse phase protein array: Validation of a novel proteomic technology and utility for analysis of primary leukemia specimens and hematopoietic stem cells. Mol Cancer Ther 5: 2512-2521.

[2] Hennessy B, et al. (2010) A technical assessment of the utility of reverse phase protein arrays for the study of the functional proteome in non-microdissected human breast cancers. Clin Proteom 6: 129-151.

[3] Smith M, Kohn R (1996) Nonparametric regression using Bayesian variable selection. J Econometrics 75: 317-343.

[4] Nott DJ, Green PJ (2004) Bayesian variable selection and the Swendsen-Wang algorithm. J Comput Graph Stat 13: 141–157.

[5] Zellner A (1986) On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In: Goel PK, Zellner A, editors, Bayesian Inference and Decision Techniques - Essays in Honor of Bruno de Finetti. Amsterdam: North-Holland, pp. 233-243.

[6] Kohn R, Smith M, Chan D (2001) Nonparametric regression using linear combinations of basis functions. Stat Comput 11: 313-322.

[7] Geiger D, Heckerman D (1994) Learning Gaussian networks. In: Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence (UAI). San Francisco, CA: Morgan Kaufmann, pp. 235-243.

[8] Madigan D, York J, Allard D (1995) Bayesian graphical models for discrete data. Int Stat Rev 63: 215-232.

[9] Denison DGT, Holmes CC, Mallick BK, Smith AFM (2002) Bayesian Methods for Nonlinear Classification and Regression. London: Wiley.

[10] Lauffenburger DA, Linderman JJ (1993) Receptors: Models for Binding, Trafficking, and Signaling. New York: Oxford University Press.

[11] Husmeier D (2003) Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. Bioinformatics 19: 2271-2282.

[12] Werhli AV, Husmeier D (2007) Reconstructing gene regulatory networks with Bayesian networks by combining expression data with multiple sources of prior knowledge. Stat Appl Genet Mol Biol 6: 15.

[13] Mukherjee S, Speed TP (2008) Network inference using informative priors. Proc Natl Acad Sci USA 105: 14313-14318.

[14] Oda K, Matsuoka Y, Funahashi A, Kitano H (2005) A comprehensive pathway map of epidermal growth factor receptor signaling. Mol Syst Biol 1: 2005.0010.

[15] Yarden Y, Sliwkowski MX (2001) Untangling the ErbB signalling network . Nat Rev Mol Cell Biol 2: 127-137.

[16] Manning BD, Tee AR, Logsdon MN, Blenis J, Cantley LC (2002) Identification of the tuberous sclerosis complex-2 tumor suppressor gene product tuberin as a target of the phosphoinositide 3-kinase/Akt pathway. Mol Cell 10: 151-162.

[17] Shaw RJ, Kosmatka M, Bardeesy N, Hurley RL, Witters LA, et al. (2004) The tumor suppressor LKB1 kinase directly activates AMP-activated kinase and regulates apoptosis in response to energy stress. Proc Natl Acad Sci USA 101: 3329-3335.

[18] Hardie DG (2004) The AMP-activated protein kinase pathway - new players upstream and downstream. J Cell Sci 117: 5479-5487.

[19] Goncharova EA, Goncharov DA, Eszterhas A, Hunter DS, Glassberg MK, et al. (2002) Tuberin regulates p70 S6 kinase activation and ribosomal protein S6 phosphorylation. a role for the TSC2 tumor suppressor gene in pulmonary lymphangioleiomyomatosis (LAM). J Biol Chem 277: 30958-30967.

[20] Yokogami K, Wakisaka S, Avruch J, Reeves SA (2000) Serine phosphorylation and maximal activation of STAT3 during CNTF signaling is mediated by the rapamycin target mTOR. Curr Biol 10: 47-50.

[21] Inoki K, Li Y, Zhu T, Wu J, Guan KL (2002) TSC2 is phosphorylated and inhibited by Akt and suppresses mTOR signalling. Nat Cell Biol 4: 648-657.

[22] Jensen CJ, Buch MB, Krag TO, Hemmings BA, Gammeltoft S, et al. (1999) 90-kDa ribosomal S6 kinase is phosphorylated and activated by 3-phosphoinositide-dependent protein kinase-1. J Biol Chem 274: 27168-27176.

[23] Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc B 58: 267-288.

[24] Friedman J, Hastie T, Tibshirani R (2010) Regularization paths for generalized linear models via coordinate descent. J Stat Soft 33: 1-22.

[25] Opgen-Rhein R, Strimmer K (2006) Using regularized dynamic correlation to infer gene dependency networks from time-series microarray data. In: Proceedings of the 4th International Workshop on Computational Systems Biology, WCSB 2006. pp. 73-76.

[26] Lèbre S (2009) Inferring dynamic genetic networks with low order independencies. Stat Appl Genet Mol Biol 8: 9.

[27] Äijö T, Lähdesmäki H (2009) Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. Bioinformatics 25: 2937-2944.

[28] Cantone I, et al. (2009) A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. Cell 137: 172-181.

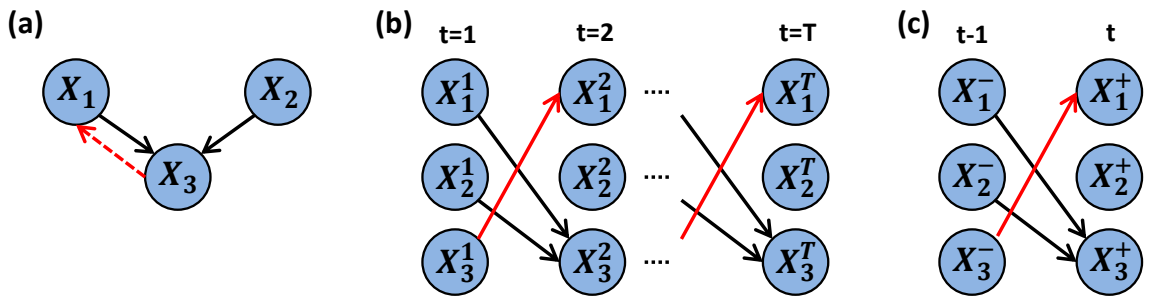

Figure S1: Dynamic Bayesian networks. (a) A (static) graph structure containing a feedback loop. This graph is not a Bayesian network (BN) because it does not satisfy the acyclicity condition. (b) The graph structure in (a) can be 'unrolled' through time to give a dynamic Bayesian network (DBN) structure in which each component is represented at multiple time points. DBNs are able to model feedback loops. (c) Due to the homogeneity of the graph structure through time, the 'unrolled' DBN in (b) can be 'collapsed' into two time slices representing adjacent time points.

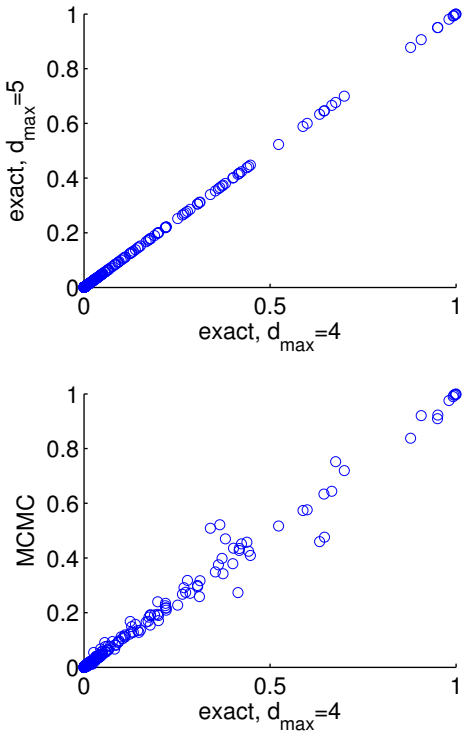

Figure S2: Effect of sparsity constraint - breast cancer cell line study. Results reported in Main Text (Figure 3a) were obtained by exact inference with a maximum in-degree of $d_{\max} = 4$. These results were compared with results obtained by exact inference with maximum in-degree increased to $d_{\max} = 5$ (top), and by Markov chain Monte Carlo-based inference without any in-degree restriction (bottom).

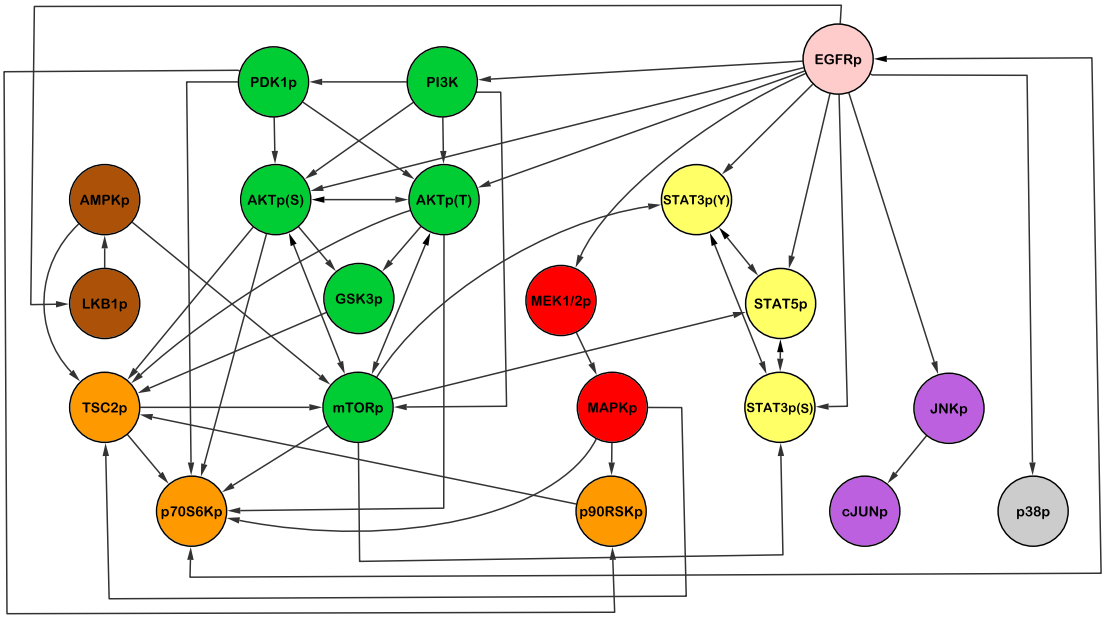

Figure S3: Network prior. Existing biology is captured and integrated during modeling using a prior probability distribution on graphs $P(G) \propto \exp(\lambda f(G))$, with $f(G) = -|E(G) \backslash E^*|$ where $E(G)$ is the set of edges contained in G and E^* is a set of *a priori* expected edges. The graph shows edge set E^*. Edges represent interactions through time. Each node also has a self-loop edge (i.e. an edge starting and finishing at the same node, these are not displayed). The edge set includes expected indirect edges which operate via components not included in our study.

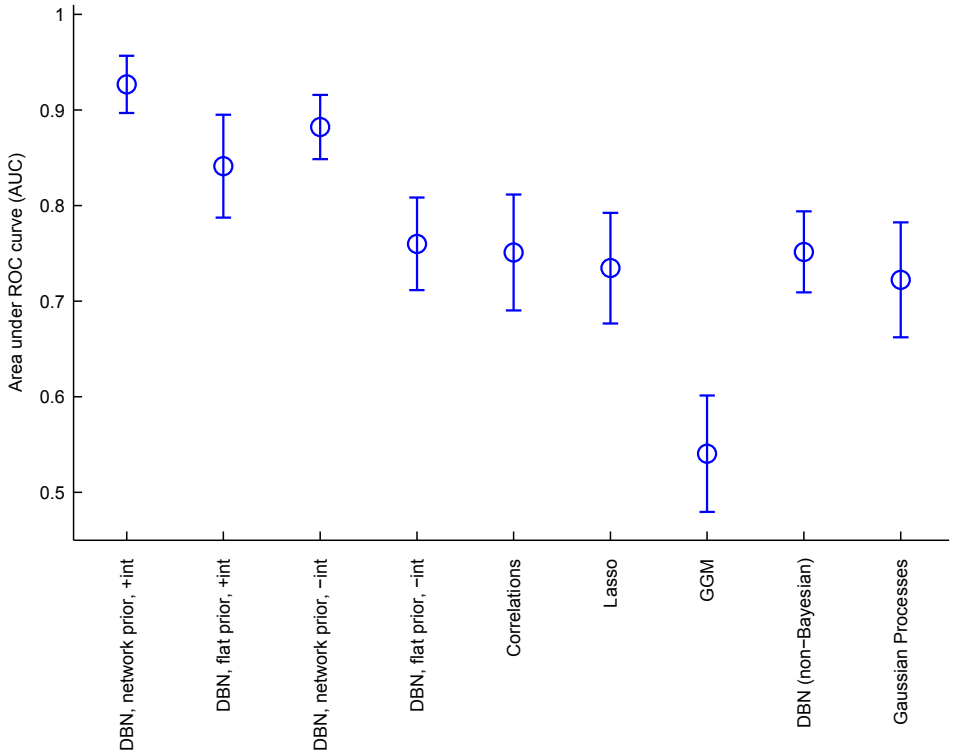

Figure S4: Simulation study. Area under the ROC curve (AUC; see Figure 2 in Main Text for average ROC curves). AUC provides a measure of accuracy in network inference; higher values indicate better performance. Simulated data, with dimensionality and sample size matching the cancer proteomic study (20 proteins, 8 time points, 4 complete time series per protein), were generated from known graph structures by ancestral sampling. Graph structures were created to be in only partial agreement with the network prior shown in Figure S3. For each data-generating graph 10 out of a total of 30 edges were not in the prior graph, while the prior graph had 54 edges that were not in the data-generating graph (see Supplementary Text for full details of simulation). Results shown are mean AUC$\pm$SD over 25 iterations. Legend - "DBN, network prior": proposed DBN inference approach using a network prior, weighted objectively by empirical Bayes; "DBN, flat prior": proposed DBN inference approach using a flat prior over network space; "+int/-int": with/without interaction terms; "correlations": absolute Pearson correlations between proteins at adjacent time points; "Lasso": ℓ_1-penalized regression; "GGM": a Gaussian graphical model approach for time series data; "DBN (non-Bayesian)": a non-Bayesian method for DBN inference; "Gaussian Processes": a non-parametric Bayesian approach using Gaussian processes. See Supplementary Text for further details of these methods.

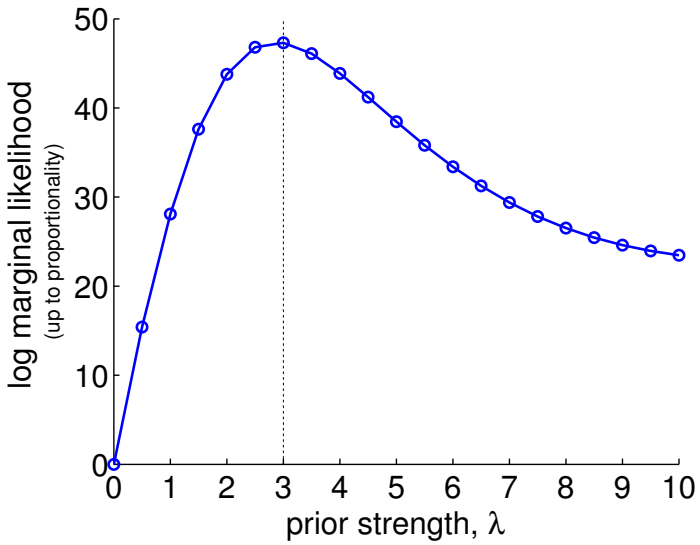

Figure S5: Objective weighting of informative network prior - breast cancer cell line study. Empirical Bayes marginal likelihood vs. prior strength λ. An informative prior on networks was used to integrate proteomic data with existing knowledge of signaling topology (derived from available signaling maps, see Main Text and Supplementary Text for details). Prior strength λ was set by an empirical Bayes approach. This was done by empirically maximizing marginal likelihood $p(\mathrm{data} \mid \lambda)$ as shown (in increments of 0.5); this gave $\lambda = 3$, the value used in all analyses.

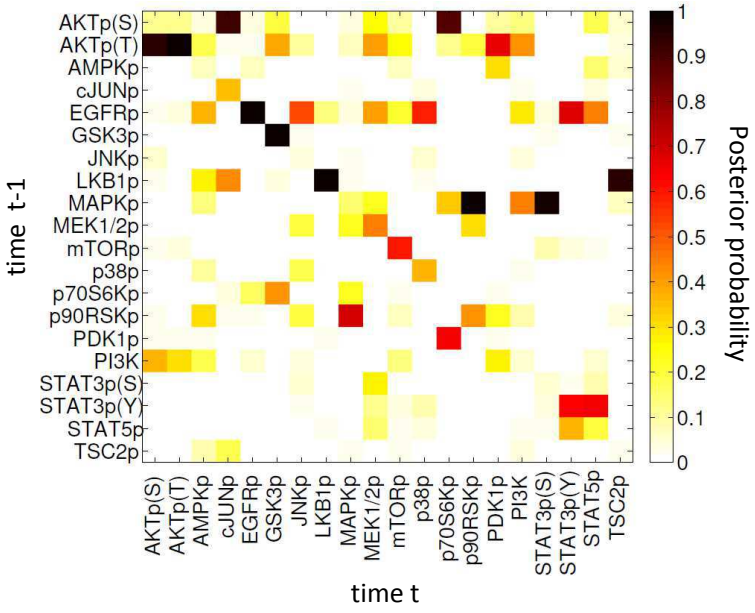

Figure S6: Posterior edge probabilities - breast cancer cell line study. Heatmap showing all 400 edge probabilities for the inferred network in Figure 3a, Main Text.

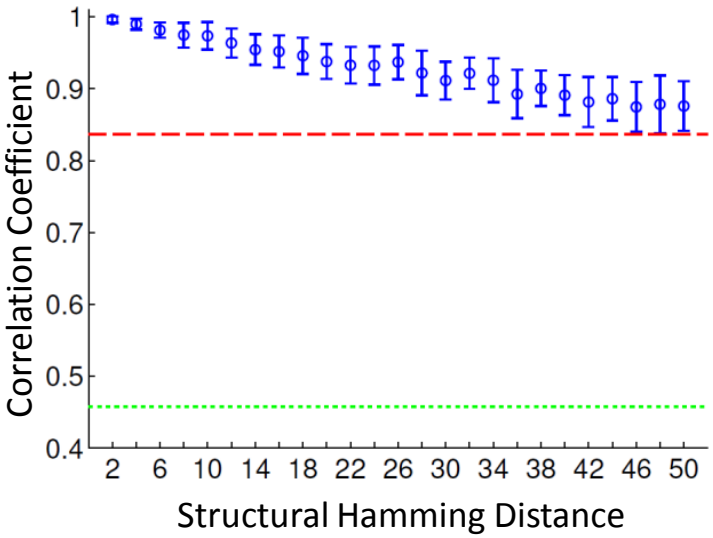

Figure S7: Sensitivity to prior graph - breast cancer cell line study. An informative prior on networks was used to integrate proteomic data with existing knowledge of signaling topology. The prior graph was perturbed and results obtained compared to those reported in Figure 3a (see Supplementary Text for details). Pearson correlation coefficients (between posterior edge probabilities) are shown as a function of number of edge changes in the prior graph ('Structural Hamming Distance'; larger values indicate a greater change to the prior graph). Dashed red line and dotted green line show the correlation between reported results (with $\lambda = 3$) and those obtained with a flat prior and prior only respectively. See also Figure 3c in Main Text.

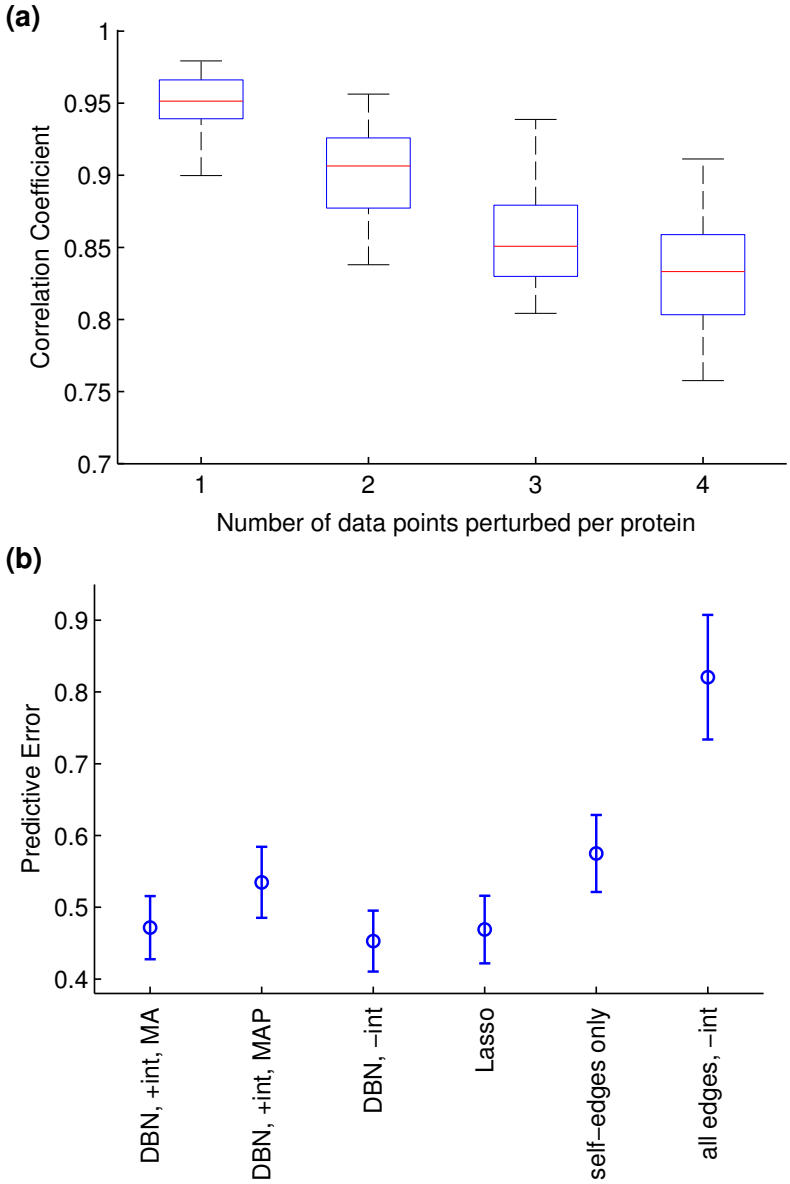

Figure S8: (a) Sensitivity to data perturbation and (b) cross-validation - breast cancer cell line study. (a) Data were removed, for each of the 20 proteins under study, from between 1 and 4 randomly selected time-point/condition combinations: this corresponded to removing between 1/32 and 1/8 of the data. Deleted data were replaced with the average of adjacent time ponts. Pearson correlation coefficients are shown between edge probabilities inferred from perturbed data and those obtained from the original, unperturbed data (Figure 3a in Main Text). Results shown are over 25 iterations (except for "1", in which all possible deletions were carried out). (b) Predictive capability was empirically assessed by leave-one-out-cross-validation. Results shown are mean absolute predictive errors $\pm$SEM for DBN network inference with interaction terms in the linear model and either exact model averaging ('DBN, +int, MA') or using the highest scoring graph ('DBN, +int, MAP); DBN network inference without interaction terms using exact model averaging ('DBN, -int'); variable selection via ℓ_1-penalized regression ('Lasso'); a baseline auto-correlative analysis ('self-edges only'); and a baseline, non-sparse linear model, with each variable predicted from all others ('all edges, -int').

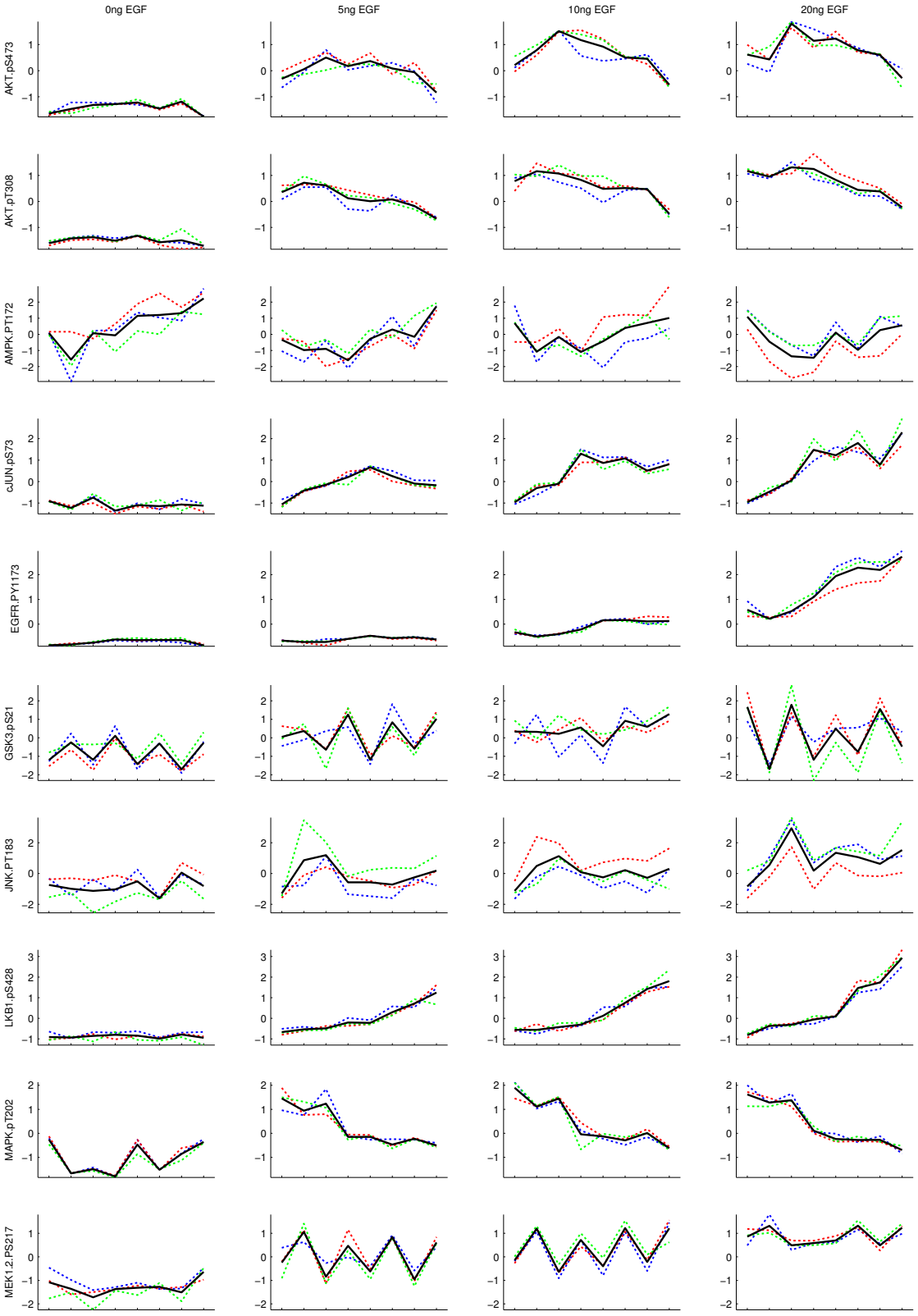

Figure S9: Time courses for MDA-MB-468. Colored lines are raw triplicates; black lines are averages. Time courses are standardized to have zero mean, unit variance across all conditions for each protein.

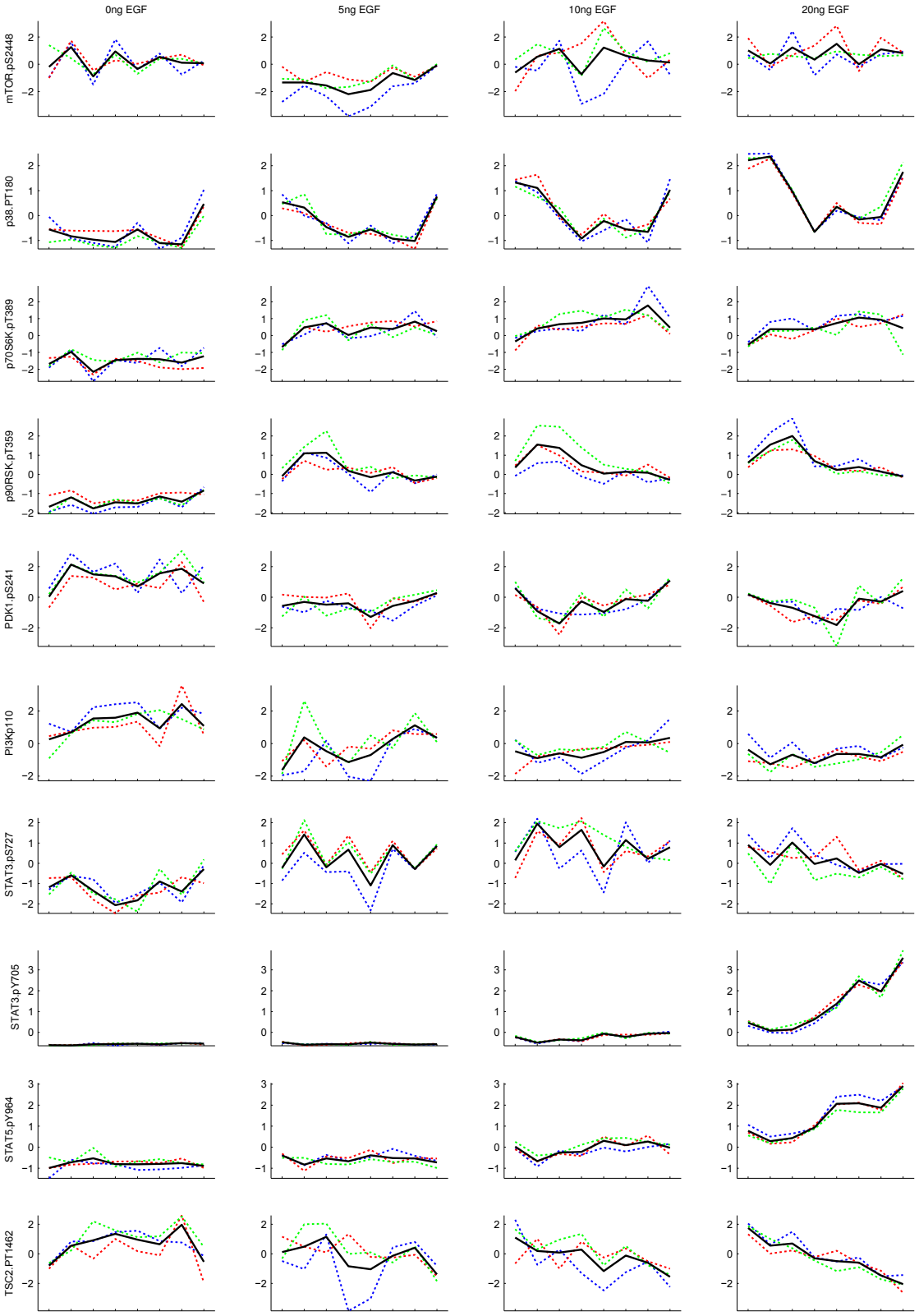

Figure S9: Time courses for MDA-MB-468.

Short Name (as used in Main Text)	**Antibody Name**	**Company**	**Catalogue**
AKTp(S)	AKT pS473	Cell Signaling	9271
AKTp(T)	AKT pT308	Cell Signaling	9275
AMPKp	AMPK pT172	Cell Signaling	2535
cJUNp	c-Jun pS73	Cell Signaling	9164
EGFRp	EGFR pY1173	Millipore	05-483
GSK3p	GSK3 pS21/9	Cell Signaling	9331
JNKp	JNK pT183 Y185	Cell Signaling	9251
LKB1p	LKB1 pS428	Cell Signaling	3051
MAPKp	MAPK pT202 Y204	Cell Signaling	9101
MEK1/2p	MEK 1/2 pS217	Cell Signaling	9121
mTORp	mTOR pS2448	Cell Signaling	2971
p38p	p38 pT180	Cell Signaling	9211
p70S6Kp	p70S6K pT389	Cell Signaling	9205
p90RSKp	p90RSK pT359	Cell Signaling	9344
PDK1p	PDK1 pS241	Cell Signaling	3061
PI3K	PI3K	Epitomics	1683
STAT3p(S)	STAT3 pS727	Cell Signaling	9134
STAT3p(Y)	STAT3 pY705	Cell Signaling	9131
STAT5p	STAT5 pY964	Cell Signaling	9351
TSC2p	TSC2 pT1462	Cell Signaling	3611

Table S1: Validated primary antibodies used in the cancer cell line study.