Supplementary methods

Method comparisons: description of methods

We compare EPoC to several other methods briefly described here.

EPoC, as described in the article, estimates the CNA-driven network G. However, the EPoC algorithm can also be applied to estimate the transcriptional network A by simply replacing the CNA data matrix ΔU with the mRNA data matrix ΔY in Step 3 of the algorithm. We denote this approach by EPoC A.

`glasso` (Friedman *et al.*, 2008) estimates the (sparse) inverse correlation matrix from a set of data. The inverse correlation matrix reflects the direct dependencies, whereas the correlation matrix includes both direct and indirect interactions. Under our model formulation, $\Delta Y \simeq -A^{-1}\Delta U$, and so the correlation matrix of the mRNA expression levels $\Sigma_{YY} = A^{-1}\Sigma_{UU}(A^{-1})^T$, or equivalently, the inverse correlation matrix $\Sigma_{YY}^{-1} = A\Sigma_{UU}^{-1}A^T$. The estimate of Σ_{YY}^{-1} thus generates an undirected version of A. Similarly, GeneNet is geared toward finding the direct links between transcripts using the inverse correlation matrix, but employs a different form of shrinkage or elimination of indirect effects from `glasso` (Opgen-Rhein & Strimmer, 2007). ARACNE uses a mutual information criterion to separate directly dependent transcripts from those dependent only through other transcripts (Margolin *et al.*, 2006). We use publicly available GeneNet and ARACNE software packages in our comparison, and our matlab/C implementation of `glasso` optimized for speed and verified to agree with the published `glasso` R-package.

Note, EPoC A can be applied to estimate the transcriptional network A using either (a) the mRNA data only, or (b) the mRNA and CNA data. Defining log-transformed and zero-centered mRNA expression and CNAs as Δy_i and Δu_i, $i = 1, \ldots, n$. For (a) we assume a transcript model $a_{ii}\Delta y_i + \sum_{j \neq i} a_{ij}\Delta y_j + r_i = 0$, where a_{ij} denote the transcriptional interactions and r_i the non-CNA related impact on transcription ("noise" in our model). That is, we regress Δy_i on other mRNA levels Δy_j for all genes $i = 1, \cdots, n$. For (b) we use model $a_{ii}\Delta y_i + \Delta u_i + \sum_{j \neq i} a_{ij}\Delta y_j + r_i = 0$, and first obtain the residuals of a regression of each mRNA transcript on its CNA (essentially on $a_{ii}\Delta y_i + \Delta u_i$), then regress these residuals on the mRNA levels of all other mRNA transcripts. The robustness performance (Figure 6A) of EPoC A is slightly worse in case (b) compared with (a). This supports that the appropriate manner in which to include CNA profiles in the network modeling of mRNA expression is through the CNA-driven network formulation (EPoC G). Since `glasso`, ARACNE and GeneNet networks are based on pairwise (partial) correlations or similar dependency measures, it is not as easy to include CNA data directly in these approaches. The cross-correlation matrix corresponding to model (b) above is not symmetric as the pairwise interaction models assume. Applying the methods directly to the residuals in (b) worsens the performance of all three methods (as expected since this does not correspond to a mechanistic model for transcription and biological information in the residuals is much reduced compared with the original mRNA data).

eQTL is a standard class of methods to associate SNPs to mRNA levels. We adapt previous work in SNP-eQTL analysis (e.g. Stranger *et al.* (2005, 2007a,b)), here replacing SNPs with CNAs. We thus apply univariate linear regression of the gene expression levels on the copy number signals, followed by calculation of the nominal p-values for the association, followed by a p-value cutoff to obtain a set of eQTL significant associations. We obtain networks of different sizes by sweeping the p-value cutoff between 10^{-16} and 1. (For our robustness tests,

described below, we note that the permutation criterion in Stranger *et al.* (2007a,b) gives a nominal p-value cutoff of around 10^{-4}).

We set up the method `remMap` as developed in Peng *et al.* (2008). This method involves several pre-processing steps. First, the CNA data is converted to CNA-intervals using fixed order clustering. Second, a set of mRNA–mRNA interactions are identified by running a method called `space`, based on a partial correlation analysis of the mRNA data (similar to `glasso`). The number of mRNA-mRNA interactions are selected using a power-law assumption (on the distribution of degree=number of links per node). In Peng *et al.* (2008), a power of 2 is deemed reasonable for the data and `space` is run to find a mRNA-mRNA network that best matches this assumption. Third, for each mRNA transcript we now build a model based on other transcripts and CNA interval data. For transcript i, mRNA transcripts j that have been identified to be linked to i using `space` are automatically included, and those that have not identified are excluded. The CNA interval that contains transcript i is also automatically included in the model. `remMap` then uses a combined $L1$ and $L2$ elastic net penalty to select a subset of the other CNA intervals to be linked to transcript i. An additional penalty encourages the same CNA interval to appear as predictors for all transcripts ("master predictors"). The method can therefore be thought of as a hybrid of EPoC A/G analyzed for genomic regions. We use the publicly available `space` and `remMap` software packages for our comparisons. In the technical comparison we apply the methods to a random subset of 500 genes. Therefore, the fixed order clustering into genomic regions is not included in the comparative analysis (since genomic location makes little less sense in a random gene set). Since this eliminates a processing step from `remMap`, the corresponding results in Figure 6A are likely optimistic (ignoring instabilities in the estimation of genomic intervals).

LirNet (Lee *et al.*, 2009) is designed to derive a transcriptional module network from combined SNP and mRNA data. Given a set of transcript clusters and a set of possible regulators, this algorithm identifies SNP and mRNA regulators for each cluster by elastic net regression (combined $L1$ and $L2$ penalties). An additional feature of the algorithm is that the $L1$ penalties can be learnt from annotation features (e.g. the position of the SNP inside the gene). We replace SNPs with CNAs and adapt the basic algorithm to apply LirNet to the TCGA data as follows; (i) we define k clusters by k-means clustering of the transcript data (Lee *et al.*, 2006); (ii) we assign the same set of possible regulators as for EPoC (see main text); (iii) we run LirNet, using the author's own software distribution, in "flat mode", for a given $L1$, $L2$ set of penalties; and, (iv) after each run, re-assign each gene to the cluster that best explains its profile. This is repeated until convergence. The above procedure thus depends on choosing the parameters k (cluster number), $L1$ and $L2$. Using $L1$ to control network size in Figure 6, we tested all possible constellations of $k = 1, 2, 3, ...$ and $L2 = 1, 2, 3, 4, 5, 10, 15, 20, ...$ For Figure 6 (network consistency and PPI matching), we obtained maximum consistency for $k = 5$, $L2 = 20$. For supplementary Figure 2 (prediction), we obtained minimum prediction error for $k = 24$ and $L2 = 4$. We note that the use of priors can improve LirNet prediction performance (Lee *et al.*, 2009). To try to compensate for the use of flat LirNet priors (i.e. not including such prior information), we modulated the consistency test (Figure 6A) so that the same initial clustering was used for the A and B datasets, which introduces a strong bias in favor of LirNet consistency, since the initial clusters are identical. While LirNet performs less well in the prediction tests than `remMap` and EPoC, we speculate that this could be compensated by using LirNet with appropriate informative priors. We defer such explorations to future work.

Sparse canonical correlation has also been put forth as an alternative non-network approach to integrating CNA and mRNA data. As an interesting side-note, while the sparse

SVD of G produces biomarker modules, where a subset of CNAs are linked to a subset of mRNAs, this is not the same as a sparse canonical correlation (CCA) of the CNA (U) and mRNA (Y) data (Waaijenborg *et al.*, 2008; Witten *et al.*, 2009). SVD of G focuses on CNA as the input or driver of mRNA changes. In contrast, sparse canonical correlation treats the mRNA and CNA data symmetrically, finding a sequence of linear combinations of a subset of CNAs that are maximally correlated with linear combinations of a subset of mRNAs. The CCA components are obtained from eigenvalue decompositions of $\Sigma_{UU}^{-1}\Sigma_{UY}\Sigma_{YY}^{-1}\Sigma_{YU}$ (input CNA) and $\Sigma_{YY}^{-1}\Sigma_{YU}\Sigma_{UU}^{-1}\Sigma_{UY}$ (output mRNA) respectively, where Σ_{UU} denotes the CNA–CNA covariance matrix, Σ_{YY} the mRNA-mRNA covariance, and $\Sigma_{UY} = \Sigma_{YU}^T$ the cross-variance between CNA and mRNA. Under our model formulation $Y = GU + \Gamma$ where, assuming U and Γ uncorrelated, $\Sigma_{YY} = G\Sigma_{UU}G^T + \Sigma_{\Gamma\Gamma}$ and $\Sigma_{UY} = \Sigma_{UU}G^T$. Here, $\Sigma_{\Gamma\Gamma}$ is the covariance of the noise term. Plugging in these expressions into the canonical correlation eigenvector problems above, it is easily seen that the canonical correlations do not generally agree with the SVD of G (eigenvectors of GG^T and G^TG). Complete agreement is possible in pathological cases, such as when both Σ_{UU} and $\Sigma_{\Gamma\Gamma}$ are factors of the identity matrix (or having eigenvectors coinciding with the right and left SVD components of G respectively). If Σ_{UU} and $\Sigma_{\Gamma\Gamma}$ differ from these special cases (e.g. identity matrices, which corresponds to completely uncorrelated CNAs and/or noise), canonical correlation and SVD of G can differ substantially. In our toy example, SVD of G correctly identifies CNA biomarkers and mRNA responders (Figure 1C). In contrast, CCA is susceptive to, and reflective of, the structure of $\Sigma_{\Gamma\Gamma}$ in either input or output components, or both. As structure in the noise term is a realistic concern in our glioblastoma data set, where $\Sigma_{\Gamma\Gamma}$ captures all the mRNA–mRNA dependencies that are non-CNA related, we do not consider CCA further in this paper, but reserve such comparisons for future work centering on network decompositions related to survival.

Method comparisons: setup and results

The setup for the comparative analysis of the methods is summarized here. We first construct two replicate versions of the TCGA dataset, A and B. A comprises array-CGH and Agilent array measurements from MSKCC; B comprises Agilent array-CGH profiles and Affymetrix U133A mRNA profiles generated at Harvard and Broad Institute, with both A and B consisting of 146 individually matched samples. For 100 iterations, we select a mixture of the 250 genes with the highest mRNA variance in one of the datasets, plus an additional random 250 genes from the 10672 genes studied. This way, we get a set of genes that can be analyzed also by the slowest methods (`remMap`, `glasso`, ARACNE), and which introduces a bias in favor of the methods that uses mRNA data only. We subsequently run each method for each of a series of parameter values corresponding to stringency (`glasso` ρ, ARACNE dpi, GeneNet significance threshold, EPoC λ, `remMaP` $L1$ penalty, `LirNet` number of clusters and $L2$ penalty, and eQTL p-value cutoff), resulting in a series of networks of different sizes. These are analyzed with respect to network agreement between datasets A and B using Kendall's W.

In Figure 6A, the robustness of network estimation using EPoC is compared with alternative methods. We use Kendall's W to compare the agreement between networks estimated on two independent data sets using EPoC G, EPoC A, `glasso`, ARACNE, GeneNet, eQTL, `remMap` and LirNet. EPoC G is clearly superior in terms of robust network estimation for most network sizes. For very large networks, `glasso` is somewhat better. GeneNet exhibits near-constant performance across the full range of network sizes, and performs the worst of the methods compared, with essentially random agreement between networks. Finally, ARACNE has a strong tendency to produce very connected networks even at maximum stringency settings, and its performance on small networks is hard to assess. The obtained

results suggest that ARACNE performs slightly better than GeneNet.

EPoC A and `remMap` exhibit very similar performance, and is overall worse than EPoC G. The reason for this is that transcriptional interactions (the A matrix of EPoC A, the `space` networks for `remMap`) are difficult to estimate due to the strong mRNA-mRNA correlations in the data. In fact, examining just the robustness of `space` we find that on average Kendall's W is 0.8. That is, $1 - W = 0.2$ of the level of the `remMap` curve in Figure 6A can be attributed to just the `space` pre-processing step, with an additional 0.2-0.3 coming from the instability of estimating the G component of `remMap`. `glasso` performs slightly better than EPoC A, but worse than EPoC G. However, as mentioned above, `glasso` is restricted to produce undirected (non-causal) networks. In addition, we find that `glasso` is much slower than EPoC A and `glasso` networks larger than 1500 genes are intractable in practice.

LirNet and eQTL perform worse still in terms of robustness. LirNet, like `remMap` suffers from instability introduced in the pre-processing step where modules are estimated. eQTL (as mentioned in Results) performs worse than EPoC largely due to being based on a univariate modeling of mRNA–CNA couplings.

Note, all methods benefit from the use of standardized mRNA amplitudes, and those results are the ones shown in Figure 6A.

From a practical stand-point, we think it valuable to note that the algorithmic speed of the methods vary greatly. In our simulation setup we use 500 genes such that even the slowest methods can be compared. For this number of genes, EPoC takes a few seconds to run (3–6s) whereas `glasso` and GeneNet take a factor 5-10 longer. Beyond 1500 genes, `glasso` takes a prohibitively long time to run. `remMap` takes on average 40–60 seconds to

estimate networks based on 500 genes and does not scale to the full data set due to memory requirements and speed. For example, on 3000 genes the `space` step alone takes more than one hour. LirNet is slightly faster than `remMap`. ARACNE takes on average 100–200 seconds to estimate networks for 500 genes and becomes prohibitively slow to run on the full data set. Finally, EPoC takes about 90 second to run on the full set of 10672 genes. (All run times were obtained on a desktop computer, Mac Pro, 2x2.8 GHz quad-core Intel Xeon).

We compare predictive performance of EPoC, `remMap` and LirNet using the same simulation setup for 500 genes as described above. These methods were chosen to compare since they produce natural prediction models as algorithmic output. For EPoC prediction we fit EPoC G and EPoC A on a training set of data, and predict using model averaging. `remMap` generates predictions in a similar fashion since the elements of the A is obtained first through `space`, and then coefficients estimates for the A and G components of prediction are obtained using elastic net (Supplementary Figure 2)

For pathway comparisons, we download Reactome, IntAct, and HPRD from Pathway-commons.org, and map identifiers to the 10672 genes in our dataset. We subsequently calculate the undirected shortest path R_{ij} for all gene pairs (i,j) in these databases using Johnson's algorithm (Johnson, 1977). For a given network G, we subsequently calculate the enrichment (relative proportion) as:

$$\frac{P(R_{ij} = k \mid i \text{ and } j \text{ are connected in G})}{P(R_{ij} = k \mid i \text{ and } j \text{ are connected in a permutation of } G_{\text{permuted}})},$$

calculated across nondiagonal elements ($i \neq j$) and where $G_{permuted}$ is generated by random permutation of the nondiagonal G elements (1000 simulations) (Figure 6B). For Figure 6, $k = 2$ is used.

Finally, we quantify the similarities of the different methods as follows. We apply and optimize each method to find a network of fixed size (here 500 connections). We then use Kendall's W to compute method-method structural similarities. We apply hierarchical clustering to these results and produce a method-dendrogram (Supplementary Figure 1). The figure shows clear structural separation between transcriptional mRNA-based networks (`glasso`, EPoC A, ARACNE and GeneNet) and the genotype-driven networks (LirNet, eQTL, `remMap` and EPoC G). Gene content comparison show a complementary set of processes identified in mRNA-based networks compared with genotype-driven networks (Supplementary Figure 1).

Supplementary experimental information: Primer sequences

GAPDH	Forward primer	5′-GAA GGT GAA GGT CGG AGT C-3′
GAPDH	Reverse primer	5′-GAA GAT GGT GAT GGG ATT TC-3′
NDN	Forward primer	5′-ACTGAGGAGTTCGTCCAAATGAAT -3’
NDN	Reverse primer	5′-TGATTTGCATCTTGGTGATTTCG -3′
CPNE8	Forward primer	5′-ACCCCTACTGTGATGGCATTGA-3′
CPNE8	Reverse primer	5′-GGGAGCCATCCTTTACAGAAGAAG-3′
FGF9	Forward primer	5′-ACCACAGCCGATTTGGCATT -3′
FGF9	Reverse primer	5′-CTTCTCATTCATCCCGAGGTAGAG -3′
KCHN8	Forward primer	5′-CCGAGAAGGTCATGAGAGTGATGT -3′
KCHN8	Reverse primer	5′-TGGGAGTCGCTTGTTGATGTTG -3′

References

Friedman J, Hastie T, *et al.* (2008) Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**: 432–41

Johnson D (1977) Efficient Algorithms for Shortest Paths in Sparse Networks. *J Acm* **24**: 1–13

Lee SI, Dudley AM, *et al.* (2009) Learning a prior on regulatory potential from eQTL data. *PLoS Genet* **5**: e1000358

Lee SI, Pe'er D, *et al.* (2006) Identifying regulatory mechanisms using individual variation reveals key role for chromatin modification. *Proc Natl Acad Sci USA* **103**: 14062–7

Margolin AA, Nemenman I, *et al.* (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* **7 Suppl 1**: S7

Opgen-Rhein R, Strimmer K (2007) From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC systems biology* **1**: 37

Peng J, Zhu J, *et al.* (2008) Regularized Multivariate Regression for Identifying Master Predictors with Application to Integrative Genomics Study of Breast Cancer. *arXiv* **stat.AP**

Stranger BE, Forrest MS, *et al.* (2005) Genome-wide associations of gene expression variation in humans. *PLoS Genet* **1**: e78

Stranger BE, Forrest MS, *et al.* (2007a) Relative impact of nucleotide and copy number variation on gene expression phenotypes. *Science* **315**: 848–53

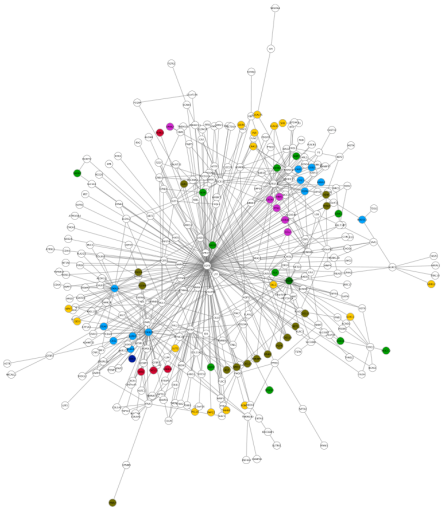
Stranger BE, Nica AC, *et al.* (2007b) Population genomics of human gene expression. *Nat Genet* **39**: 1217–24

Waaijenborg S, de Witt Hamer PCV, *et al.* (2008) Quantifying the association between gene expressions and DNA-markers by penalized canonical correlation analysis. *Stat Appl Genet Mol Biol* **7**: Article3

Witten DM, Tibshirani R, *et al.* (2009) A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics* **10**: 515–34

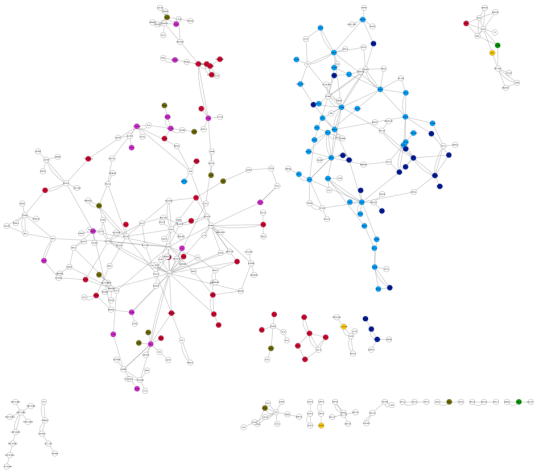
Supplementary figure 1

A. EPoC CNA-driven network



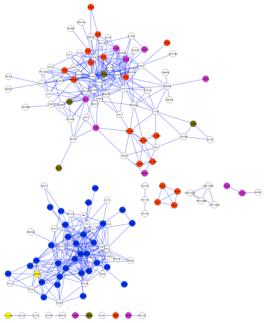
cell differentiation (GO:0030154)
nervous system development (GO:0007399)
cell-cell signaling (GO:0007267)

B. EPoC, transcriptional network



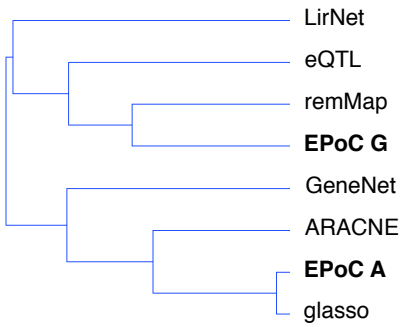
inflammatory response (GO:0006954)
immune response (GO:0006955)
cell cycle (go: GO:0007049)

C. glasso network



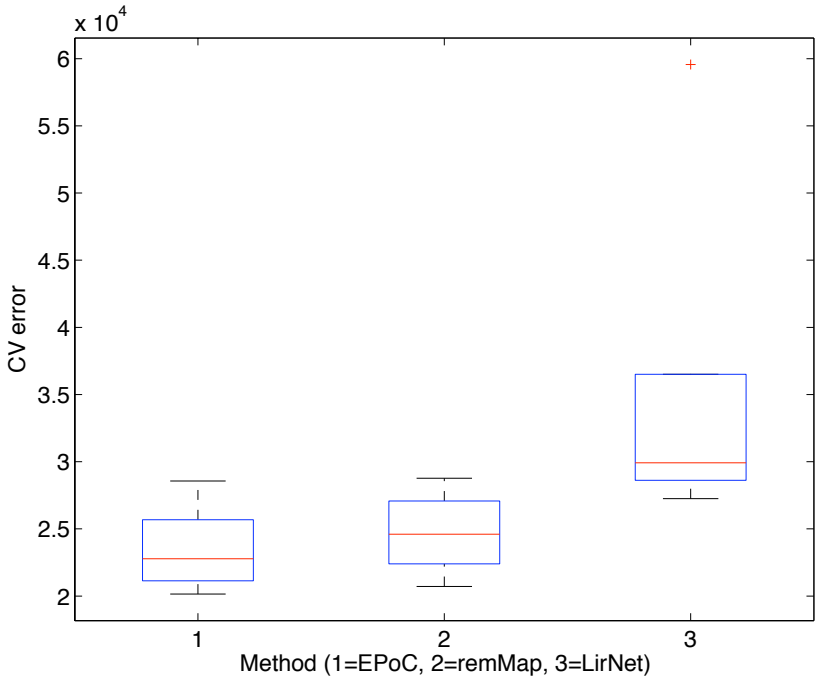
immune response (GO:0006955)
cell cycle (go: GO:0007049)

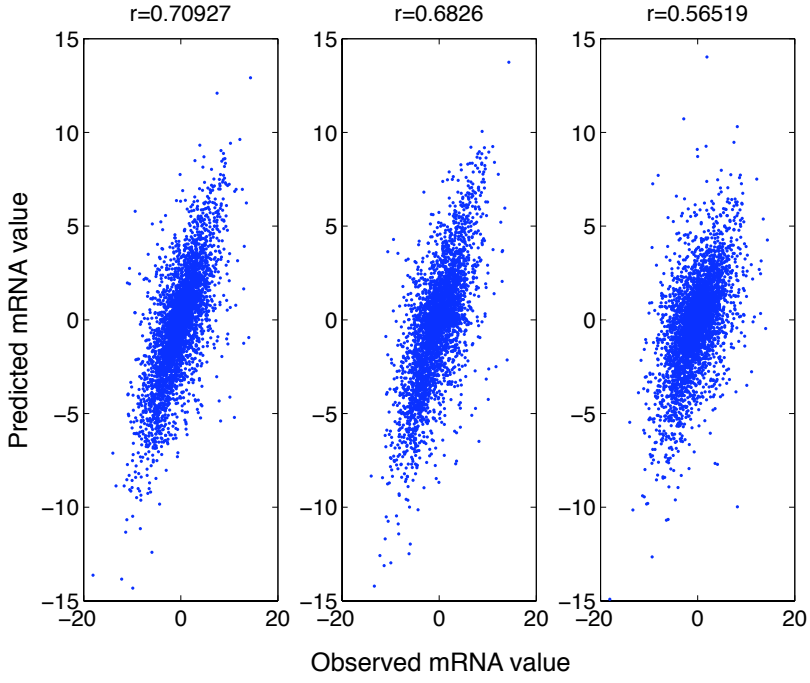
D. network similarity



Supplementary figure 1: Differences in gene content between EPoC G and A networks. Comparison of top the CNA-driven (A), transcriptional (B) and `glasso` (C) networks. Top three enriched GO process terms highlighted (the corrected Fisher's test p-values ($< 10^{-9}$ for all terms shown) are used as a ranking principle and not as evidence of network links). In the CNA-driven network, we detect numerous genes involved in cell-cell signaling, and developmental processes, whereas the transcriptional network contains a large number of associated with inflammatory and cell cycle associated processes. The `glasso` method, which shows robustness results that are comparable to EPoC (Figure 6A) produces a solution which has similar gene content to the EPoC A solution (lower left). (The `glasso` network is computed from a random subset of 2000 genes, since this method does not scale to 10000+ genes). (D) Hierarchical clustering of network solutions (single linkage, 1-fractional network overlap as distance); note that EPoC A networks group with transcriptional network methods (ARACNE and GeneNet) and EPoC G groups with `remMap` and similar genotype-based methods.

Supplementary figure 2





Supplementary figure 2: Prediction performance. The results of 10-fold cross-validation test of EPoC, `remMap` and LirNet with respect to prediction of mRNA levels from mRNA and CNA features selected by each methods (Methods). Each box shows the range of prediction errors across each of the 10 simulations, demonstrating similar performance of the three methods on the TCGA glioblastoma data. The simulation was done on 500 gene subsets due to speed limitations of `remMap` (Methods). As an example, scatter plots of measured (X axis) and predicted (Y axis) mRNA levels, with Pearson correlation coefficient r at a similar level. Each method was optimized with respect to lasso ($L1$) and ridge ($L2$) parameters. LirNet, which is based on transcriptional modules, was also optimized for different module numbers for optimal performance.