Methods

PIIKA methodology

The steps performed by PIIKA (Fig. 1) are described in detail here. The description can be used, for example, to perform the steps of PIIKA independently of the software discussed earlier. PIIKA is implemented in the R programming language (*1*), with accessory scripts written in bash (that is, a UNIX or LINUX shell) or Perl (see Equipment). Specific R packages used in PIIKA are mentioned wherever used and can be obtained from the locations described in Equipment. Individual steps are illustrated by data samples as appropriate. In the data samples, an initial row and initial column with informative labels have been added for explanatory purposes. These may differ from the actual content of the header row and header column internally associated with the matrix by R.

Input to PIIKA

As described in the Instructions, the "create_combined_file.pl" script is used to combine the data from each individual array into the format accepted by the main PIIKA script ("piika.R"). The file produced by "create_combined_file.pl" has the format exemplified by:

peptide	protein	T1R1F	T1R1B	T1R2F	T1R2B	T2R1F	T2R1B	T2R2F	T2R2B
FAK_Y397	Q05397	33057	31091	31021	29946	43192	41861	30947	30593
FAK_Y397	Q05397	32571	31415	35434	34411	47452	46250	30716	30259
FAK_Y397	Q05397	37917	35868	44621	43545	44635	42990	31370	31069
4E-BP1_T37	Q13541	24342	30439	29591	32692	39270	42323	29800	31511
4E-BP1_T37	Q13541	25266	29416	32329	37331	37824	41222	29550	31091
4E-BP1_T37	Q13541	35696	37934	38773	43347	39216	41473	33299	34486
APE1_S289	P27695	34449	32072	29519	28403	49454	43819	32833	31121
APE1_S289	P27695	37955	35687	33782	32482	53944	48349	31895	31304
APE1_S289	P27695	35627	32936	42191	40318	45903	40279	33362	31808

where "T*i*R*j*F" stands for "Treatment *i* Replicate *j* Foreground" and "T*i*R*j*B" stands for "Treatment *i* Replicate *j* Background." In this example, the first three rows of numeric values provide the spot information for three intra-array replicates of the peptide phosphorylation location FAK_Y397 across two replicates (R1 and R2) and two treatments (T1 and T2). The remaining rows, in groups of three, contain the spot information for "4E-BP1_T37" and "APE1_S289."

Data processing before analysis

1. Background subtraction is performed on the input data. For each row and each pair of columns recording intensity values, the background intensity is subtracted from the foreground intensity. A new table is created with the results. As an example, the following matrix is the result after background subtraction is performed on the data example above:

peptide	T1R1	T1R2	T2R1	T2R2
FAK Y397	1966	1075	1331	354
FAK Y397	1156	1023	1202	457
FAK Y397	2049	1076	1645	301
4E-BP1 T37	-6097	-3101	-3053	-1711
4E-BP1 T37	-4150	-5002	-3398	-1541

4E-BP1 T37	-2238	-4574	-2257	-1187
APE1 S289	2377	1116	5635	1712
APE1 S289	2268	1300	5595	591
APE1 S289	2691	1873	5624	1554

In the initial row we have added for explanatory purposes "T*i*R*j*", which stands for "Treatment *i* Replicate *j*". As before, each group of three rows of numeric values provides the spot information for the three intra-array replicates (across two replicates and two treatments).

2. The resulting data is transformed with a variance stabilization (VSN) model (*2*). The transformation calibrates all of the data to a positive scale while maintaining the structure within the data and alleviating variance-versus-mean dependence.

Note: The latter problem occurs when the variances of signal intensities for individual peptides are not constant, but increase as mean intensity increases (figs. S1, S2, and S3). Correction of the problem is necessary because subsequent statistical tests assume a constant variance. In addition, the data from various arrays are brought to the same scale by VSN to enable comparisons between subjects, treatments, etc. The R function "vsn2" from the vsn *package is used for the transformation. It is designed for data in a table in which a single column corresponds to all of the data from a single physical microarray. This was the motivation for having intra-array replicates on separate rows in the input to this step. The wrapper function "justvsn", which is also from the* vsn *package, is used to simplify the use of "vsn2."*

3. If there are intra-array replicates (multiple spots for individual peptides on a single array), the matrix is rearranged to have each row contain all of the replicates of a unique peptide. This is necessary because the remainder of the methodology assumes that each row of the matrix contains all replicates for a given peptide, including intra-array replicates. For example, suppose there are three intra-array replicates per peptide and that the data input to this step, after VSN transformation, are as follows:

peptide	T1R1	T1R2
FAK Y397	11.508	11.357
FAK Y397	11.162	11.333
FAK Y397	11.541	11.358
4E-BP1 T37	8.157	9.113
4E-BP1 T37	8.690	8.423
4E-BP1 T37	9.426	8.557
APE1 S289	11.665	11.376
APE1 S289	11.624	11.459
APE1 S289	11.777	11.699

where the header row and column (with informative labels) have been added for explanatory purposes and "T*i*R*j*" stands for "Treatment *i* Replicate *j*". "Replicate" here could be either an inter-array or biological replicate. The dataset is then rearranged to give:

peptide	T1R1I1	T1R1I2	T1R1I3	T1R2I1	T1R2I2	T1R2I3
FAK Y397	11.508	11.162	11.541	11.357	11.333	11.358
4E-BP1 T37	8.157	8.690	9.426	9.113	8.423	8.557
APE1 S289	11.665	11.624	11.777	11.376	11.459	11.699

where "T*i*R*j*I*k*" stands for "Treatment *i* Replicate *j* Intra-array Replicate *k*".

Note: No averaging of values is performed in steps 1 to 3. This is to maximize the number of replicates for subsequent statistical tests (χ^2 test, F *test, and* t *test). Only in subsequent analysis, such as the clustering analysis in step 10, is the average for each of the peptides in a single treatment taken over the transformed replicate intensities.*

Note: For sites that undergo little or no phosphorylation in a given experiment, it is not uncommon for the area surrounding a spot to undergo greater staining than the spot itself because of nonspecific interactions between the stain and the glass. This results in background intensities that are greater than the foreground intensities. Fortunately, the subsequent negative values do not present problems for the software pipeline because of the VSN transformation.

4. A χ^2 test is used to examine the variability for each peptide across technical replicate spots; that is, replicates on the same chip or multiple chips for the same subject under the same treatment (*3*). The results of the χ^2 tests are stored in a matrix with rows corresponding to those of the dataset. In later steps (for example, step 8), peptides with statistically significant variability may be explicitly eliminated from the dataset. For each peptide, the null hypothesis H_0 claims that there is no difference among intensities from the technical replicate spots, and the alternative hypothesis H_A states that statistically significant variation exists among them. The χ^2 test statistic (TS_1) is as follows:

$$TS_1 = \frac{(n-1)s^2}{\hat{\sigma}^2} \qquad (1)$$

where n is the number of technical replicates for each peptide in the treatment, $s^2 = (1/n)\sum_{i=1}^{n}(y_i - \bar{y})^2$ is the sample variance of the technical replicates for each peptide in a treatment, $\hat{\sigma}^2 = (1/M)\sum_{j=1}^{M} s_j^2$ is the mean of all the variances for the technical replicates of the M peptides in the treatment (that is, the total number of distinct peptides included in an array), and:

$$P \text{ value} = P[TS_1 > \chi^2(n-1)].$$

The peptides with P values less than a threshold are considered to have an inconsistent pattern of phosphorylation across the technical replicates and may be eliminated in subsequent steps (steps 8, 10, or 11). When this is done, a strict confidence level (that is, 0.01) is used so that as much information as possible is retained. That is, peptides with statistically significant P values are

eliminated, so the more stringent the threshold, the fewer are discarded. The *P* values are calculated with the R function "pchisq."

Note: If there are multiple technical replicate arrays, then the χ^2-test is performed for all of the replicates for a given treatment, giving a P *value for that treatment. If there are multiple biological replicate arrays, then the χ^2-test is performed separately for each array corresponding to a given treatment, and the* P *value for that treatment is the minimum* P *value among all these arrays.*

5. One treatment may be the biological control for another treatment. Subtraction of the biological control may be useful to prepare the data for downstream analysis, such as clustering based on differences in the extent of phosphorylation. Therefore, if desired, the intensities induced by the treatments can be adjusted by subtracting the intensities of the corresponding controls. If there are multiple subjects, the biological control of the same subject is used. For example, given the following row of control and treatment information for peptide P1 in a dataset:

	BCI1	BCI2	BCI3	T1I1	T1I2	T1I3	T2I1	T2I2	T2I3
P1	4.67	3.85	4.47	3.76	4.52	3.42	4.26	4.30	4.02

this operation yields:

	T1I1'	T1I2'	T1I3'	T2I1'	T2I2'	T2I3'
P1	-0.91	0.67	-1.05	-0.41	0.45	-0.45

where there is a single control (BC) for two treatments (T1 and T2), "BCI*j*" stands for "Biological Control Intra-array Replicate *j*", "T*i*I*j*" stands for "Treatment *i* Intra-array Replicate *j*", and "T*i*I*j*'" stands for "adjusted Treatment *i* Intra-array Replicate *j*". Thus BCI1 is subtracted from T1I1 and T2I1 to yield T1I1' and T2I1', respectively. As before, an initial row and initial column with informative labels have been added to the matrix values for explanatory purposes.

6. For each of the peptides, an *F* test is used to determine whether there are statistically significant differences among the subjects under the same treatment condition (*4*). This step is only applied to datasets in which there are biological replicates, and where there is a concern of variation across subjects. For example, the *F* test may be important for experiments involving outbred species, including humans, where variability in responses across individuals is common. Data for peptides determined to be inconsistently phosphorylated may be eliminated in subsequent analysis (for example, in step 8). Because subtraction of the biological background may affect subject-subject variability, this step is performed after step 5.

For a given peptide, let a be the number of subjects, n the number of intra-array replicates, N the total number of replicates for each treatment, and μ_i the mean response in the i^{th} subject for each treatment. The null hypothesis H_0 claims that $\mu_1 = \mu_2 = \ldots = \mu_a$, or the mean phosphorylation

intensities elicited by the peptide among the subjects are the same, and the alternative hypothesis H_A states that not all subject means are equal. The F-statistic (TS_2) is calculated as:

$$TS_2 = \frac{MS_B}{MS_W} \tag{2}$$

where,

$$MS_B = \frac{SS_B}{df_B} = \frac{\sum_{i=1}^{a} n(\bar{y}_i - \bar{y})^2}{a-1} \qquad \text{(Mean Squared Between Subjects)}$$

$$MS_W = \frac{SS_W}{df_W} = \frac{\sum_{i=1}^{a}\sum_{m=1}^{n} (y_{im} - \bar{y}_i)^2}{N-a} \qquad \text{(Mean Squared Within Subjects).}$$

Above, $\bar{y}_i \equiv \hat{\mu}_i$ is the sample mean for the i^{th} subject, $\bar{y} \equiv \hat{\mu}$ is the grand mean for all of the subjects, and y_{im} is the individual response of the m^{th} replicate in the i^{th} subject. Finally,

$$P \text{ value} = P[TS_2 > F(a-1, N-a)]$$

For a given treatment, any peptide with a *P* value less than a threshold for any subject is considered inconsistently phosphorylated among the subjects and may be eliminated from subsequent analysis (for example, in step 10). As with step 4, a strict confidence level (such as 0.01) is used so that as much information as possible is retained. The above calculations can be performed in R with the "aov" function.

7. For all peptides, one-sided paired *t* tests are used to compare their signal intensities under two conditions, for example a treatment and a control condition (*4*). This is done for all treatment-control or treatment-treatment combinations of interest. The goal is to identify those peptides for which the signal intensities are truly different under alternate conditions; that is, those peptides that are differentially phosphorylated. The paired *t* test is carried out by the function "t.test" that is built into R.

Formally, the t-test statistic (TS_3) is calculated as:

$$TS_3 = \frac{\bar{D}}{S_D / \sqrt{N}} \tag{3}$$

where $\bar{D}$ is the mean of the differences between responses for a given peptide induced by two different treatments, N is the number of differences, and S_D is their standard deviation.

Finally:

$$P \text{ value (phosphorylation)} = P\,[TS_3 > t(N-1)]$$

and

$$P \text{ value (dephosphorylation)} = P\,[TS_3 < -t(N-1)].$$

Thus, each peptide has two *P* values, one associated with the peptide being differentially phosphorylated and the other with the peptide being differentially dephosphorylated. The peptides with *P* values less than a threshold are considered differentially (de)phosphorylated. To identify as many differentially (de)phosphorylated peptides as possible, no adjustment (as for multiple hypothesis testing) is made to the *P* value, and a liberal threshold (for example, 0.1) may be used. In equation 3, *N* is the number of replicates per treatment. For example, if only one array was created for a single subject and there are three intra-array replicates, then $N = 3$. If there are three inter-array replicates and one subject, then $N = 9$, because there are 3 intra-array replicates per array. Finally, if there are 3 subjects and one array per subject with 3 intra-array replicates per array, then *N* is again 9.

Note: A paired t *test, rather than an unpaired* t *test, is used here because, for a given peptide, a particular intra-array replicate for one treatment has a corresponding intra-array replicate (in the same "block" on the array) in another treatment.*

Note: For some threshold T*, if the* P *value for differential phosphorylation of a peptide is less than* T*, then the* P *value for dephosphorylation must be greater than* T*, and vice versa.*

Note: The t *test is able to account for the variability among the replicates so that replicates with statistically significant* P *values from the* χ^2 *tests will automatically have insignificant* P *values from the* t *test. However, this does not apply to datasets with multiple subjects, because significant variation for the same peptide among the subjects under the same treatment condition might be biologically meaningful, and it may confound the analysis if these peptides are treated as if they came from the same source. This was the primary motivation for the* F *test in step 6.*

Note: The decision to not make a P *value adjustment for multiple hypothesis testing is further discussed in Future Work.*

8. The results of previous statistical tests are applied, and peptides that are differentially phosphorylated between a pair of treatments are reported. For a peptide to be deemed differentially phosphorylated, two conditions must be met. First, the *P* value of the peptide from the χ^2 test must be greater than the threshold given in step 4 for both treatments. Further, if the F test in step 6 was also applied, then the *P* value of the peptide from that test must be greater than the corresponding threshold for each treatment. That is, a peptide with a χ^2 test or *F* test *P* value less than the corresponding threshold is not reported as differentially phosphorylated because it is deemed inconsistently phosphorylated across technical replicates or among the subjects, respectively. The second condition for a peptide to be considered differentially phosphorylated is that its *P* value from either of the paired *t* tests for the treatment pair is less than the threshold given in step 7.

The strictness of the thresholds for the χ^2 test, *F* test, and *t* test has a direct effect on the number of peptides that are reported as differentially phosphorylated. There are fewer biomolecules represented on a kinome microarray (for example, 300) than are represented on a transcription microarray (for example, 30,000). Therefore, the thresholds are chosen so that a greater proportion of biomolecules are reported as statistically significantly different than might be the case with transcription microarrays. For the *t* test, peptides with statistically significant *P* values are reported, so a higher threshold yields more results. Hence, no adjustment (as for multiple hypothesis testing) is made to the *P* values, and a liberal threshold (for example, 0.1) is used. For the χ^2 test and *F* test, peptides with significant *P* values are eliminated, so the more stringent the threshold, the fewer are discarded. Therefore, a strict confidence level (for example, 0.01) is used. In step 7, individual t-tests can be performed in parallel for various pairings of treatments. It is therefore possible that a peptide has a statistically nonsignificant *P* value for one pair of treatments, but a significant *P* value for another.

Note: The results of the χ^2 test and F *test are used to eliminate peptides as candidates for being differentially phosphorylated. A peptide is only eliminated if it is inconsistent for one (or both) of the treatments involved in the* t *test. If it is only inconsistent for other treatments, then it is not eliminated.*

Note: It is possible for a peptide to have a nonsignificant P *value for a t test for a particular comparison between two treatments because of inconsistent intensity values, but for another combination of treatments, the intensity values are more consistent and the peptide has a significant* P *value.*

Note: A statistically significant χ^2 test P *value results from a large variability across replicates. This variability also results in insignificant* P *values in the* t *test. Hence, application of the χ^2 test results is not strictly necessary to categorize the peptide as being differentially phosphorylated, and can be bypassed for simplicity or efficiency reasons.*

9. The results from the treatment-treatment variability analysis in step 7 (that is, the *P* values for phosphorylation or dephosphorylation of each peptide) are reported in step 8. If there is only one treatment and a control, this often suffices for identification of differential phosphorylation. However, if there are multiple treatments relative to a single control, or multiple treatments each relative to its own control, then more complex patterns of phosphorylation may be present. For these situations, visualization of differential analysis results can facilitate the identification of patterns of differential phosphorylation across treatments.

PIIKA makes use of a simple but effective visualization paradigm. Each peptide is represented by one small colored circle that is partitioned into two sectors (semi-circles), each of which represents a different pair of comparison treatments. For example, the left sector might be a first treatment compared with its control, whereas the right sector represents a second treatment compared to its control. A label under each circle identifies the index of the corresponding peptide in the data set. The depths of the coloration in red and green in a given sector are inversely related to the corresponding *P* values for phosphorylation and dephosphorylation, respectively. For example, if the *P* value for phosphorylation is 0.001, then the redness in

percentage will be 100% × (1 − 0.001) = 99.9%. The same encoding is applied to dephosphorylated peptides and the extent of greenness. Thus, the combined color depths of red and green represent the phosphorylation status of each peptide in the microarray.

The colored circles are laid out in blocks, top-to-bottom in graphical output produced by R. The first block contains peptides differentially phosphorylated in both pairs of treatments. Below that is a block of peptides differentially dephosphorylated in both pairs of treatments. Next are two sets of peptides in which one pair of treatments exhibits increased phosphorylation and the other exhibits decreased phosphorylation. Finally, peptides with inconsistent phosphorylation (as determined by the χ^2 test in step 4 or the *F* test in step 6) are represented. Within the blocks in which the peptides are differentially phosphorylated in both pairs of treatments, the peptides with the most significant *P* values on average for phosphorylation or dephosphorylation over the treatments being compared are presented first (going left to right and then top to bottom), followed by the less statistically significant ones. Similarly, in the blocks in which one treatment results in increased phosphorylation whereas the other yields decreased phosphorylation, peptides with the largest differences between the *P* values from the treatment pairs are presented first, followed by the peptides with smaller differences. An example of the visualization for two treatment pairs is given in Fig. 2. The visualizations are generated with the R functions *"*plot*"* (to initialize the plot), "rgb*"* (for coloration), and *"*polygon*"* (to draw sectors at specific coordinates to represent treatments).

Note: The color encoding of a circle representing a peptide is specific to the treatment pairs under consideration. For example, suppose there are three treatments, a, b, and c, as well as a control, and that the spot intensities are inconsistent across subjects for treatment a, but consistent for the others. If treatments b and c versus a common control are being shown in the visualization, then the fact that treatment a is inconsistent (for this peptide) is not shown in the visualization.

Note: It is possible to distinguish between inconsistent phosphorylation across technical replicates (the result of the χ^2 test) and inconsistent phosphorylation across subjects (the result of the F *test) in the visualization. For example, the former can be rendered in white, whereas the latter can be represented in gray. The implementation in R of such a scheme is straightforward.*

Note: With more sophisticated R code, it is possible to arrange the circles in the visualization to reflect the physical layout of the array. An example is given in Fig. 3. The peptides are grouped according to "phosphorylated for all three treatment pairs", "peptides dephosphorylated for all three treatment pairs", etc. However, now the blocks of peptides are arranged left to right and then top to bottom.

Note: It is also possible to represent more than two pairs of treatments in the visualization. In general, t treatment pairs can be represented by dividing the colored circle into t sectors. An example with t = 3 is given in Fig. 3.

10. To further expose patterns in the kinome data, transformed peptide phosphorylation intensities are subjected to hierarchical clustering and principal component analysis (PCA). The aim is to cluster peptide response profiles across treatments or subject-treatment combinations.

First, however, peptides with inconsistent intensities across technical or biological replicates are removed. Such inconsistent intensities are indicated by the *P* values determined in the previous spot-spot and subject-subject variability analyses (steps 4 and 6, respectively). The same thresholds as described in step 8 are used. As opposed to the filtering in step 8, however, a peptide is removed from consideration if it is inconsistently phosphorylated for any treatment or any subject. The clustering and PCA can be across treatments or subject-treatment combinations. An average intensity is taken over the technical replicates for each treatment or subject-treatment combination. The averaged data with or without biological control subtractions is then subjected to hierarchical clustering and PCA. The dendrograms from the hierarchical clustering are augmented by heatmaps showing the averaged phosphorylation or dephosphorylation intensities.

For hierarchical clustering, three popular combinations of linkage method and distance measurement are implemented, namely "Average Linkage + (1 - Pearson Correlation)", "Complete Linkage + Euclidean Distance", and "McQuitty + (1 - Pearson Correlation)" (*5-8*). In general, each subject (or treatment) vector is considered as a singleton (that is, a cluster with a single element) at the initial stage of the clustering. The two most similar clusters are merged, and the distances between the newly merged clusters and the remaining clusters are updated iteratively. The calculations of similarity or distance between the clusters and the update step are algorithm-specific. The "Average Linkage + (1 - Pearson Correlation)" method is used by Eisen *et al*. (*9*). It takes the average over the merged (that is, the most correlated) kinome profiles and updates the distances between the merged cluster and the other clusters by recalculating the Pearson correlations between them. In "Complete Linkage + Euclidean Distance", the distance between any two clusters is considered as the Euclidean distance between the two farthest data points in the two clusters (*7, 8*). Finally, the McQuitty method updates the distance between the two clusters in such a way that upon merging clusters C_X and C_Y into a new cluster C_{XY}, the distance between C_{XY} and each of the remaining clusters, say C_R, is calculated taking into account the sizes of C_X and C_Y (*6*). These clustering methods can all be achieved with the R function "heatmap.2" from the gplots package. Input to this function includes the filtered, averaged, VSN-transformed intensity values. A particular clustering technique is specified by the arguments to the "heatmap.2" function call.

Note: The hierarchical clustering is augmented by a heatmap, which is also generated using the R function "heatmap.2". The function converts the intensity values to statistical z*-scores, and then the* z*-scores are encoded as color (green or red) intensities. Green usually means a value lower than the mean, whereas red represents a greater value.*

Note: PCA is a variable reduction procedure. The calculation is essentially a singular value decomposition of the centered and scaled data matrix (*10*)*. As a result, PCA transforms a number of possibly correlated variables into a smaller number of uncorrelated or orthogonal variables (that is, principal components). The first principal component accounts for the most variability in the data, and each succeeding component accounts for as much of the remaining variability as is possible. Usually, the first three components account for more than 50% of the variability in the data, and can be used as a set of the most important coordinates in a 3D plot to reveal the structure of the information.*

Note: The R function "prcomp" is used for PCA. A 3D plot for the PCA using the first three principal components is produced by the R function "scatterplot3d" from the package scatterplot3d*. A 2D PCA plot can be produced with the "plot" function. An example of the latter can be found in fig. S4.*

11. Although not technically part of PIIKA itself, we present here the methodology used to take output from PIIKA (the identities of the differentially phosphorylated peptides and the extent of their differential phosphorylation relative to that of peptides under control conditions) and use it to interrogate InnateDB (www.innatedb.ca) to discover known signaling pathways that are specifically influenced by the treatment under investigation (*11-15*). Typically, such a search requires the UniProt or GeneSymbol identifiers of the differentially phosphorylated peptides. These are readily available from the information about the kinome array, and are part of the input to PIIKA (see Materials).

InnateDB requires fold-change (FC) values as input, with optional *P* values, whereas the PIIKA methodology generates differences of transformed intensities and *P* values. Therefore, to use InnateDB, the differences between the VSN-transformed intensities under the control condition and a particular treatment (or between two different treatments) are converted to ratios (that is, FC values). The formula for the VSN transformation is complex, and an inverse function is not obvious. However, an important component of the VSN transformation is calculation of a logarithm to the base 2. Hence, the conversion from the transformed intensity to the FC ratio is approximated by an exponential function (anti-logarithm).

Peptides that show statistically significant subject-subject variability in the F-test in step 6 are removed with the threshold described in that step. In addition, peptides may be removed, if desired, based on the results of the χ^2 test; the threshold from step 4 is applied. Then, for a given treatment, the replicate transformed intensity values for each peptide are averaged. If the treatments under consideration are treatment and control, the averaging process yields $average_{treatment}$ and $average_{control}$, respectively, for each peptide. The fold-change for each peptide is then calculated as 2^d where $d = average_{treatment} - average_{control}$. This overall procedure converts the VSN-transformed values to FC ratios.

For each of the remaining peptides in the dataset, the following is input to InnateDB: the accession number of the protein containing the peptide representing a phosphorylation site, the synthetic FC value, and a *P* value from the one-sided *t* test. If a peptide has a positive calculated FC value, then the *P* value associated with phosphorylation is chosen. Otherwise, the *P* value associated with dephosphorylation is chosen. The protein accession number was part of the information initially input to PIIKA (see Materials). If multiple peptides come from the same protein, then the protein will appear multiple times, with an individual *P* value and FC value each time. InnateDB ignores column headers if given. A sample of input is given below:

```
protein       p-value             fold-change
Q05397        0.415457634         -1.044452633
Q13541        0.336302927         1.064849163
Q13541        0.193882405         -1.162705187
```

```
P27695          0.098999126          1.638849167
```

In the sample, there are two entries for protein Q13541, the first because of the peptide with ID 4E-BP1_T37 and the second because of the peptide 4E-BP1_T46.

Pathway analysis through InnateDB involves an interactive interface that enables specification of both *P* value and FC thresholds. These thresholds specify the user's confidence in the data set and resulting pathways. InnateDB eliminates from its analysis all peptides with a *P* value greater than the former threshold, or an FC value less in absolute value than the latter threshold. It is recommended that the FC threshold be set to a nonselective value, such as 1. This value is nonselective because the synthetic FC values will all be ≥ 1 or ≤ -1. This nonselectivity is a deliberate choice. Because the *P* value is a calculation of how statistically significant the difference is between treatments, it is the preferred basis for determining whether a peptide should be included, rather than relying on FC. It is also recommended that the *P* value threshold parameter to InnateDB be set to a liberal value such as 0.1. A more restrictive value such as 0.01 can be used, but this tends to result in very few results being reported.

InnateDB produces an extensive amount of output. The fields relevant to this analysis methodology are: (i) the pathways identified from the input proteins; (ii) the number of input proteins associated with each identified pathway; (iii) the gene symbols of the input proteins associated with each identified pathway; and (iv) a *P* value for each pathway, based on the number of proteins (corresponding to input peptides) present for that pathway. Within the web interface provided by InnateDB, identified pathways can be visualized with the Cerebral plugin (*16*) for the Cytoscape interaction viewer (*17*). The resultant visualizations can be downloaded to the user's computer. Examples of the resultant network visualizations are given (Fig. 4).

Note: As in step 9, for a particular peptide it is possible for there to be statistically significant subject-subject variability (as determined by the F *test) only for treatments not under consideration. In such a case, the peptide would not be eliminated from the analysis.*

Note: Peptides with large variability across their replicates will have statistically insignificant P *values in the* t *test (due to a large denominator in the* t *statistic), and hence will be automatically removed as a result of the threshold specified to InnateDB. Large variability across technical replicates will also result in statistically significant* P *values from the* χ^2 *test. This is one of the reasons that filtering based on* χ^2 *test results is optional in this step.*

Additional General Notes

The organization of the input data matrix, and the restriction to disallow both inter-array replicates and biological replicates, are designed to ease the analysis in R. Alternate organizations are possible, and the restriction can be eliminated if the user is willing to devote additional R code to matrix indexing operations.

Test statistics and *P* values are calculated in steps 4 and 6 for the purposes of filtering data from the analysis; however, no data are removed at those steps. Data removal is left to the subsequent steps 8 through 11. The main motivation for this is that it makes the R code for working with the

matrices easier; the loops simply iterate over all 300 peptides without the need to consider exceptions. Fortunately, the presence of the inconsistently phosphorylated peptides (the peptides that would otherwise be filtered) does not harm any of the individual statistical analyses. The second reason for this design is so that each downstream step can use the results of the statistical steps in easily customizable ways. For example, in step 8, filtering based on the χ^2-test results can be optionally performed without affecting the filtering in any other step.

The visualization of step 9 automatically deals with any inconsistencies in spot intensity. Peptides with large subject-subject variability are explicitly color-coded in white. On the other hand, peptides that have large variation across replicates will have statistically insignificant P values and hence tend to be automatically colored in brown (combined red and green). For the clustering analysis, however, these peptides need to be removed because there is no procedure that takes into account the inconsistent extent of their phosphorylation. Hence, for clustering analysis, they must be eliminated explicitly.

PIIKA is easily modified to provide information with which to search databases other than InnateDB for the discovery of known signaling pathways influenced by the treatment under investigation (step 10). For example, the Kyoto Encyclopedia of Genes and Genomes (KEGG) (www.genome.jp/kegg) (*12-14*) could be searched. The type and format of the information will be database-specific.

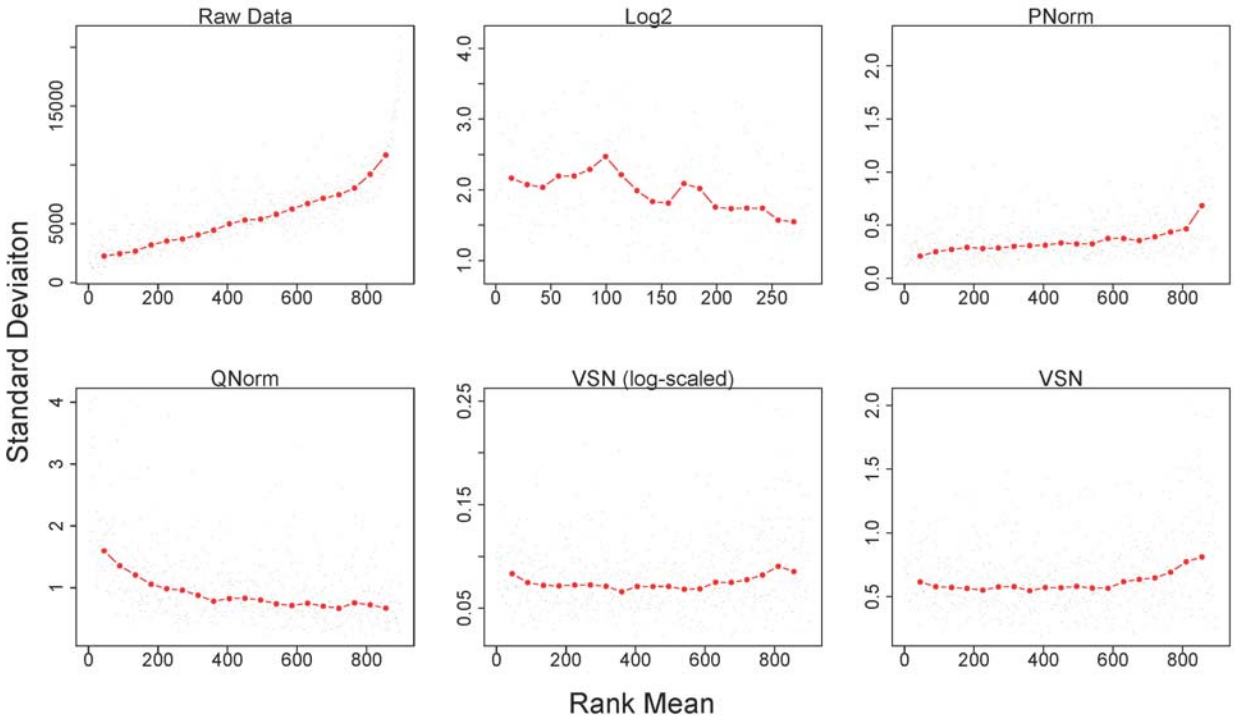

Fig. S1. Variance versus mean dependence plots before ("Raw Data") and after normalization by $\log_2$ ("Log2"), percentile normalization ("PNorm"), quantile normalization ("QNorm"), and transformation by variance stabilization ("VSN") with or without $\log_2$ scaling for the combined datasets in the case study. The rank of the mean signal intensities was plotted against the standard deviation (SD) of the corresponding peptide intensities (represented by black dots). The red dots depict the running median estimator (window-width 10%). If there is no variance-mean dependence, then the line formed by the red dots should be approximately horizontal. "Log2" refers to a simple $\log_2$ function applied after the negative values that resulted from background corrections were eliminated. The $\log_2$ function is an inbuilt function of R, and the plot was generated by the R function "meanSdPlot" from the *vsn* package (*18*).

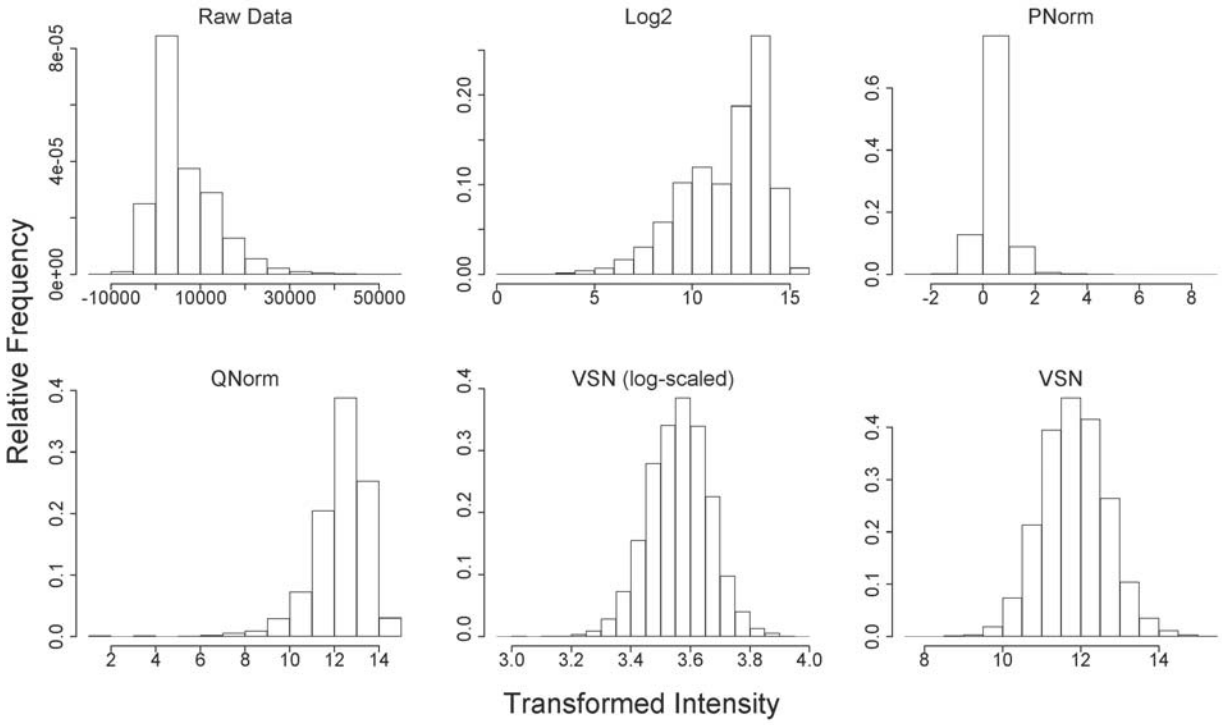

Fig. S2. Histograms of relative frequencies versus intensity before ("Raw Data") and after normalization by $\log_2$, PNorm, QNorm, or VSN with or without $\log_2$ scaling for the combined datasets in the case study. Transformations are shown as for fig. S1. For the "Raw Data" plot, the y-axis is actual frequency.

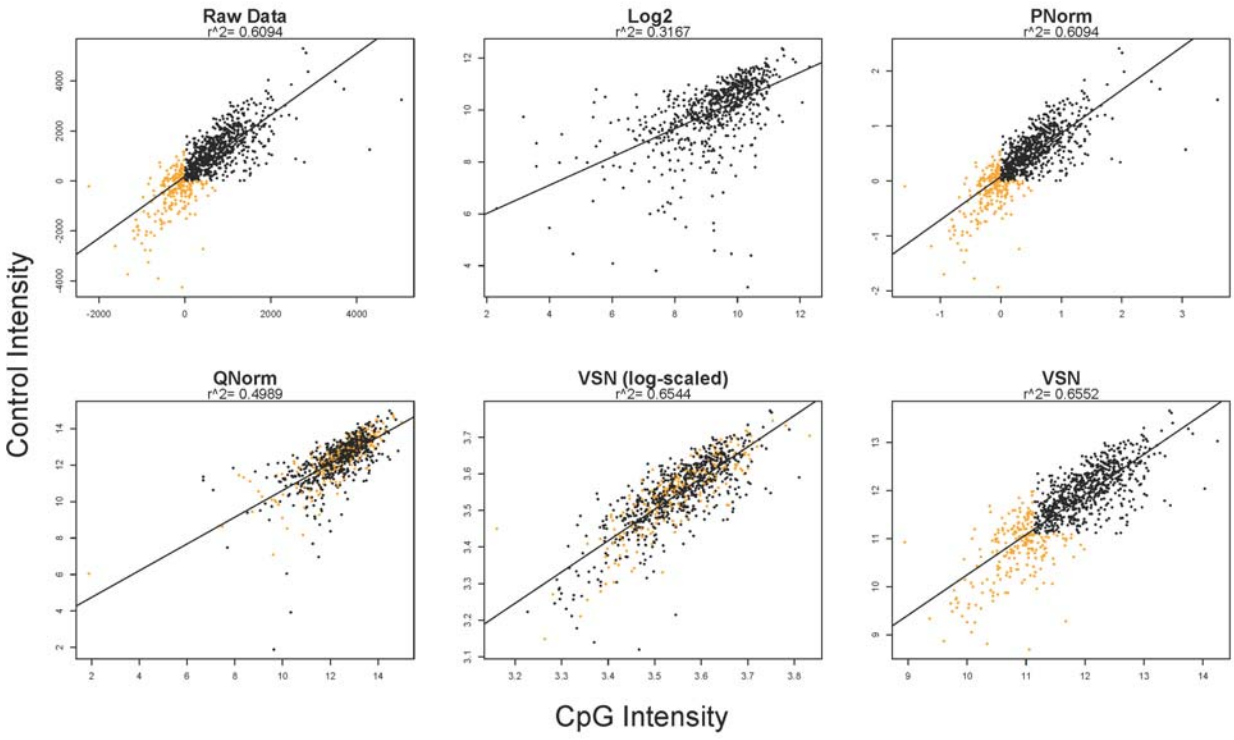

Fig. S3. Scatter plots of the signal intensities for monocytes treated with CpG oligonucleotides against the corresponding intensities from control cells treated with medium alone. The raw data were preprocessed in the following ways, as indicated: none, $\log_2$ of the positive intensities (discarding the negative ones), PNorm, QNorm, VSN (log-scaled), and VSN alone. The black and orange dots in each plot represent signal intensities after background subtraction and averaging across intra-slide replicates. If the resulting intensity for either treatment (CpG or MonoCpG) is negative, an orange dot is used. Otherwise the average intensity for both treatments is positive, and the dot is colored black. The coefficient of determination (r^2) is indicated below the title of each plot.

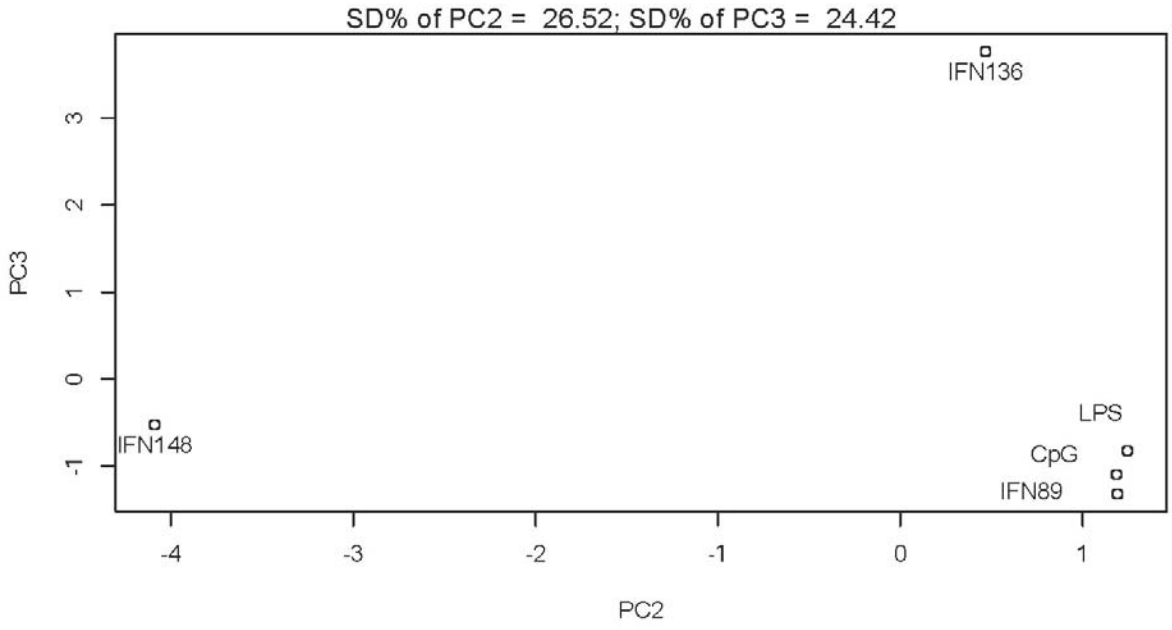

Fig. S4. Results from principal component analysis (PCA) on the intensity values from the case study. Intensity values from the three datasets were processed with our proposed PIIKA data analysis pipeline, including subtraction of biological controls, and PCA was performed on the resultant values. The second and third principal components were used for the 2D plot. The percentages of the total variability that the two PCs account for ("SD%") are displayed on the top of the plot. The data points are labeled with treatments; that is, CpG, LPS, or IFN. For the experiment involving treatment with IFN-γ, the treatment name is followed by an animal code. The R functions "prcomp" and "plot" were used for the PCA and the 2D plot, respectively.

References

1. R Development Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2009). ISBN 3-900051-07-0.
2. W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, M. Vingron, Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics* **18** Suppl 1, S96-S104 (2002).
3. S. Drăghici, *Data analysis tools for DNA microarrays* (Chapman & Hall/CRC, Boca Raton, Fla, 2003).
4. D. C. Montgomery, *Design and analysis of experiments* (Wiley, Hoboken, NJ, 2009), 7th edn.
5. K. Pearson, Mathematical contributions to the theory of evolution. III. Regression, heredity and panmixia. *Philos Trans Royal Soc London Ser A* **187**, 253-318 (1896).
6. L. L. McQuitty, Similarity Analysis by Reciprocal Pairs for Discrete and Continuous Data. *Educational and Psychological Measurement* **26**, 825-831 (1966).
7. B. Everitt, *Cluster Analysis* (Heinemann Educ., London, 1974).
8. J. A. Hartigan, *Clustering Algorithms* (Wiley, New York, 1975).
9. M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A* **95**, 14863-14868 (1998).
10. K. V. Mardia, J. T. Kent, J. M. Bibby, *Multivariate analysis* (Academic Press, London, 1979).
11. D. J. Lynn, G. L. Winsor, C. Chan, N. Richard, M. R. Laird, A. Barsky, J. L. Gardy, F. M. Roche, T. H. Chan, N. Shah, R. Lo, M. Naseer, J. Que, M. Yau, M. Acab, D. Tulpan, M. D. Whiteside, A. Chikatamarla, B. Mah, T. Munzner, K. Hokamp, R. E. Hancock, F. S. Brinkman, InnateDB: facilitating systems-level analyses of the mammalian innate immune response. *Mol Syst Biol* **4**, 218 (2008).
12. M. Kanehisa, S. Goto, KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res* **28**, 27-30 (2000).
13. M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, M. Hirakawa, From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res* **34**, D354-D357 (2006).
14. M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, M. Hirakawa, KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Res* **38**, D355-D360 (2010).
15. R. J. Arsenault, S. Jalal, L. A. Babiuk, A. Potter, P. J. Griebel, S. Napper, Kinome analysis of Toll-like receptor signaling in bovine monocytes. *J Recept Signal Transduct Res* **29**, 299-311 (2009).
16. A. Barsky, J. L. Gardy, R. E. W. Hancock, T. Munzner, Cerebral: a Cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. *Bioinformatics* **23**, 1040-1042 (2007).
17. P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, T. Ideker, Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* **13**, 2498-2504 (2003).
18. W. Huber, A. von Heydebreck, H. Sueltmann, A. Poustka, M. Vingron, Parameter estimation for the calibration and variance stabilization of microarray data. *Stat Appl Genet Mol Biol* **2**, Article3 (2003).