Matthew Ziegler
Supervised by Dr. Mark Craven
Lab section 605
Tuesday, April 19 2011

Mathematical Models of Circadian Rhythms in Gene Expression Data, and the Regulatory Role of *Mop3* in Mice

Abstract

Many genes are expressed in circadian (daily) rhythms. Identification and analysis of circadian rhythms is difficult, because gene expression time series data is often noisy and sparse. Circadian rhythms can be found using a combination of spectral analysis and harmonic regression, and when implemented as the ARSER algorithm, we find 2,423 genes in mice which appear to have circadian rhythms (at $q <= 0.05$). One gene that is believed to have an important regulatory role in circadian rhythms in mammals is *Mop3*. When the *Mop3* gene is disabled in mice, far fewer genes appear to have periodic behavior. An ensemble clustering method is used to cluster the parameters of the sinusoids produced by ARSER, producing groups of genes with appear to be regulated similarly, allowing us to learn about the regulatory networks behind circadian rhythms in mammals.

Introduction

Many organisms exhibit behaviors that are repeated at a regular interval. Behaviors that follow a daily pattern are called circadian rhythms. Some examples of circadian rhythms are animals sleeping every night, daily raising and lowering of body temperature, and opening and closing of plant stomata on leaves.

Many circadian rhythms are not controlled by external stimuli, but are instead controlled by internal regulatory mechanisms inside the cell (though external stimuli often still play a role in the behavior.) It has been observed that many genes are expressed following a circadian rhythm, and are highly expressed at certain times of day but less highly expressed at others (Smith et al., 2008.)

Complex regulatory networks for circadian rhythms have been shown to exist inside cells. A gene called *Mop3* has been identified as playing a key role in regulating circadian rhythms in mammals (Bunger et al., 2000.) When the *Mop3* gene is disabled in mice, the circadian rhythms for many genes change or disappear, though there is still much to be learned about the specific workings of *Mop3* (Akashi et al., 2005.)

In one experiment, mice without functional *Mop3* genes were bred, and their behavior was monitored over a period of several days. When deprived of external temporal cues such as light-dark cycles, mice lost circadian patterns in their wheel-running activities, and exhibited rhythms with periods less than 24 hours (Bunger et al., 2000). The absence of a functional *Mop3* gene has been shown to alter the temporal expression of many other genes, indicating that many other genes are regulated by *Mop3* (Smith et al., 2009).

For our experiment, we detect genes in mice that appear to have periodic behavior. We then cluster the cycling genes into groups of genes that have similar sinusoids in order to make inferences about their regulation, and cluster the cycling genes based upon the way in which their behavior changes between when the *Mop3* gene is knocked out.

Gene expression measurement

Relative levels of gene expression can be measured using RNA microarray technology. The amount of RNA for a particular gene within a cell is used as a proxy for measuring that gene's expression level. A microarray is a chip which has been embedded with short RNA sequences corresponding to particular genes (Fig. 1). RNA from a cell is florescently tagged, and the RNA is washed over the microarray chip. Cellular RNA binds to the complimentary RNA strand on the chip, and then the amount of cellular RNA corresponding to each gene can be read off of the chip using the fluorescent tags (Fig. 2).

Microarray data is often challenging to work with because it tends to be noisy. Gene expression levels vary between cells, and the variation can obfuscate the signals present in the data. This problem can be somewhat mitigated by pooling samples from multiple organisms, and running them on a single microarray chip to get an average reading between the samples.

To analyze changes in expression levels over time, it is necessary to create a time series of microarray data, where microarray readings are taken over a period of time at set time intervals. Because microarray experiments are currently expensive, microarray time series tend to be sparse and cover short periods of time. They are nevertheless widely used because they can measure expression levels for thousands of genes at once.

Period detection

There are several different approaches for identification of circadian rhythms in microarray time series, consisting mainly of time-domain methods, which analyze a signal with respect to time, and frequency-domain methods, which analyze a signal with respect to frequency. (Chudova et al., 2009) The most widely-used is an algorithm called COSOPT, which is a time-domain pattern-matching method that finds circadian rhythms by measuring the goodness-of-fit between the time series and a set of cosine curves (Straume, 2004). Pattern-matching algorithms are efficient to compute, but often fail to find patterns that are not sinusoidal (Chudova et al., 2009).

Frequency-domain approaches to circadian rhythm detection rely on spectral analysis. Signals can be decomposed into their component frequencies. The component frequencies of a signal are called the signal's spectrum. For example, the spectrum of a musical chord is the frequencies of the individual notes that make up the chord.

One frequency-domain method is Fisher's G-test, which computes a periodogram of the experimental data. It then uses the G test to determine the significance of the dominant frequency. Fisher's G-test and other frequency-domain approaches are robust to noise and can detect a wide variety of patterns, including patterns that are not sinusoidal. They are, however, constrained by the low frequency resolution of short microarray time series, and are often difficult for biologists without math background to understand (Yang and Su, 2010).

A newer algorithm called ARSER has been recently introduced by Rendong Yang and Zhen Su (2010), which combines the time-domain and frequency-domain approaches (Fig. 3). ARSER first computes a linear regression and removes the linear trend from the data. The detrended data is smoothed by a fourth-order Savitzky-Golay filter, which removes spectral pseudo-peaks created by noise.

To identify the periods of each profile, ARSER then calculates the autoregressive (AR) spectrum of the data. An AR model of order *p* is fit the time series using:

$$x_t = \sum_{i=1}^{p} \alpha_i x_{t-i} + \epsilon_t$$

where ϵ_t is white noise and α_i are model parameters with $\alpha_p \neq 0$ for an order *p* process. After computing the AR model, ARSER estimates the spectrum of the model using the model parameters instead of the original data:

$$p_x(\omega) = \frac{\sigma_\epsilon^2}{\left|1 + \sum_{k=1}^{p} \alpha_k e^{-i\omega k}\right|^2} \quad 0 \leq \omega < \pi$$

Where σ_ϵ^2 is the variance of white noise, and α_k are the AR model parameters. Periods in the gene expression time series will appear as peaks in the AR spectrum. If there is no periodicity in the data, there will not be significant peaks in the spectrum.

After identifying genes which exhibit periods, ARSER uses the periods found in the AR step to compute a harmonic regression for the time series using:

$$x_t = \mu + \sum_{i=1}^{n} \beta_i \cos(2\pi\ f_i t + \Phi_i) + \epsilon_t$$

where x_t is the observed value at time *t*, μ is the mean of the time series, β_i is the amplitude, Φ_i is the phase, ϵ_t are outliers that are unrelated to the cycles, f_i are the dominant frequencies found in the AR spectrum, and *t* are the sampling times. The harmonic regression models the waveform present in the data.

ARSER returns the periods computed for each gene using the AR model, and the parameters of the sinusoid fitted by the harmonic regression. It also returns *p* and *q* values describing the confidence of ARSER's estimation of the period, and an R^2 value representing the goodness-of-fit of the harmonic regression.

ARSER is able to identify non-sinusoidal patterns in genes using AR spectral analysis, and computes a sinusoid for every pattern. It has shown to be more sensitive than COSOPT and Fisher's G-test in finding circadian rhythms in an *Arabidopsis* gene set (Yang and Su, 2010). An important caveat of the ARSER algorithm is that it can only use data where the time points are evenly-spaced.

Clustering algorithms

Given the large sizes of microarray datasets, consisting of tens of thousands of genes, it is necessary to employ clustering algorithms to look for order in the data. Clustering algorithms look for order in a dataset by placing data points into groups, so that the members of each group are similar to each other.

There are many different clustering algorithms in use today, which exploit different mathematical properties of the data to form clusters, and form clusters with different properties. Some are algorithms more suited to different types of data than others, and they often produce different results. It is often difficult to determine whether a clustering represents meaningful order in the data, or is arbitrary output of the algorithm. One technique used to evaluate clusterings is ensemble clustering, where several different clustering algorithms are used on the same dataset, and the results of the algorithms are compared to each other for similarity (Coen 2010.)

For many clustering algorithms, it is necessary to define a distance between each pair of data points. In this paper we will use the Euclidean distance metric, as defined by:

$$D_{i,j}=\sqrt{\sum_{x=1}^{N}(i_x-j_x)^2}$$

where *i* and *j* are the vectors representing the two data points, *N* is the number of dimensions in *i* and *j*, and $D_{i,j}$ is the distance between *i* and *j*. The Euclidean metric represents the shortest geometric distance between the two points.

K-means clustering is a simple and widely-used clustering algorithm. In k-means clustering, k number of points are randomly placed on the same manifold as the data points. The k points represent the center-points of k clusters. The k-means algorithm iteratively moves the center points around the manifold to heuristically minimize the distance between each center-point and the distances of the data points that are assigned to each cluster. The algorithm stops moving the center points when some stopping criterion is met.

There are two steps to each iteration of the k-means algorithm. First, each data point is assigned to its nearest center point, forming clusters. Secondly, the center points of the cluster are re-computed to be the means of all of the points in each cluster, so the average distance between each data point and its center point is minimized. The iterations are stopped upon convergence, when the the amount of movement of the cluster center points is below some threshold, or alternatively after a set number of iterations.

The k-means algorithm is simple and has a linear complexity, so it is fast and efficient to compute. One caveat is that the number of clusters must be specified before the algorithm is run. The algorithm tends to produced circle-shaped clusters, where all of the data points are about the same distance away from the cluster center. Another caveat is that the initial placement of the cluster center points has an effect upon the final results. To mitigate this problem, the algorithm is usually run several times with different, randomly-placed starting points, and the best clustering is used.

Another simple and widely-used clustering algorithm is agglomerative hierarchical clustering. Initially, each data point is its own cluster. At each iteration of the algorithm, the closest two clusters are merged together until there is only one cluster. There are several methods for determining the distance between two clusters; we will use the average-link method, defined as:

$$D_{c_u,c_v}=avg\{D_{a,b}\,|\,a\in c_u, b\in c_v\}$$

where c_v and c_u are two clusters, *a* and *b* are points within c_u and c_v, respectively, and D_{c_u,c_v} is the distance between clusters c_u and c_v.

Agglomerative hierarchical clustering produces a tree-like structure, showing the relationships between each of the clusters. The structure can be visualized using a dendrogram, where the heights are measurements of the distances between each cluster. A tree-like clustering can be turned into a "flat" clustering by drawing a line at any hight of the dendrogram, and using the clusters that intersect it.

Evaluation of clusterings

It is often difficult to determine whether a clustering is truly representative of order within the data, or if it is only arbitrary output of the algorithm. In this experiment we will measure the "stability" of the clustering to assess the goodness of our clusterings.

When we vary some aspect of the clustering, if the new clustering is highly similar to the original clustering (it is stable,) that indicates that the clustering is more likely to be representative of order present in the dataset (Ben-Hur, 2002). There are a variety of methods available for measuring the similarity of clusterings. We will use CDistance, developed by Mike Coen.

CDistance is a geometric method of assessing cluster similarity, which takes into account the spatial data behind each cluster (as opposed to a set-theory based approach, which only takes into account the assignment of points to clusters.) The CDistance approach frames the problem of assessing cluster similarity as the Optimal Transportation Distance problem, which asks *what is the cheapest way to move a set of masses from souces to sinks, which are some distance away?* We are given two weighted points sets (A, p) and (B, q) in the metric space (Ω, d_Ω) , where p and q are the sets containing the weights of points in A and B, respectively. The optimal solution to the problem is computed as:

$$d_{OT}(A, B; p, q, d_\Omega) = \sum_{i=1}^{|A|} \sum_{j=1}^{|B|} f^*_{i,j} d_\Omega(a_i, b_j)$$

where $f^*_{i,j}$ is the optimal flow between (A, p) and (B, q), and is computed using a linear program. The naive solution to the problem is given by:

$$d_{NT}(A, B; p, q, d_\Omega) = \sum_{i=1}^{|A|} \sum_{j=1}^{|B|} p_i q_j d_\Omega(a_i, b_j).$$

The “similarity distance,” that we will use to measure the similarity of our clusters is measured using the relationship between the optimal transportation distance and the naive transportation distance, where our two clusterings are represented by (A, p) and (B, q). The similarity distance is given by:

$$d_s(A, B; p, q, d_\Omega) = \frac{d_{OT}(A, B; p, q, d_\Omega)}{d_{NT}(A, B; p, q, d_\Omega)}.$$

Similarity distance is zero when the data points in each clustering perfectly overlap, and approaches one as the distance between them increases. To harness the similarity distance to measure the distance between clusterings *A* and *B* of datasets *D* and *E* in the metric space (Ω, d_Ω), we define the "clustering distance" as:

$$d(A,B)=d_s(A,B,\pi,p,d'_{OT})$$

where the weights $\pi=(|\alpha|/|D|:\alpha\in A)$ and $p=(|\beta|/|E|:\beta\in B)$ are proportional to the number of points in the clusters, and where the distance d'_{OT} is the optimal transportation distance between clusters $\alpha\in A$ and $\beta\in B$ with uniform weights.

The stability of our clustering can be evaluated by clustering subsamples of the full dataset. If the clusterings produced by the subsamples are similar, as measured by the CDistance algorithm, it is likely that our clustering represents some true order within the data.

Gene ontology

The gene ontology is a collaborative project to annotate genes with a controlled vocabulary across species (Blake et al., 2011). Known genes are labeled by researchers according to their cellular component, molecular function, and biological process. There are tools available to search for labels which have a high frequency in a given set of genes, relative to all annotated genes for that organism.

Gene ontology queries can be useful for evaluating clusterings of genes. If there are different frequencies of terms between clusters, that indicates that there is a true difference between the two clusters, and that the clustering describes true order within the data.

Rationale

A better understanding of the regulation of circadian rhythms in mice will help us to understand the regulation of circadian rhythms in humans. Disruption of circadian rhythms is an important factor in many sleep disorders, and is sometimes caused as a side effect of medications. A better understanding of the genetic regulation of circadian rhythms may help us understand and mitigate these human health problems.

To our knowledge, there are no available clustering algorithms that are designed to specifically cluster sinusoids, though researchers from many fields have used a myriad of clustering algorithms to address the problem. Research on the effectiveness of different algorithms for clustering sinusoids is beneficial to many different fields that make use of signal analysis.

Methods

Gene expression measurement

Data from the Chris Bradfield oncology lab was used to analyze *Mop3*'s effects upon circadian rhythms in mice. The dataset represents two groups of mice: one group of wild type mice with a functional *Mop3* gene, and one group of mice with the *Mop3* gene “knocked out” in their liver cells, so they had no functional *Mop3* gene in their livers.

Over a period of 3 days in the wild type mice and 2 days in the knockout mice, 3 mice were sacrificed from each group every 4 hours. Samples were taken from their livers, and RNA from the three mice were pooled to create a single sample, which was measured using a microarray chip. In total, 24,034 genes that were measured had significant expression on at least 75% of the microarrays.

Two-color hybridized microarray chips were used for the experiment. To provide a reference point, RNA from all of the mice across all time points were pooled and tagged red. The separate RNA samples for each time point were tagged green. The two pools of RNA were applied to the microarray simultaniously, and the ratio between the green expression level and red expression levels were measured. The data set that we use for finding circadian rhythms contains the base-10 log for the green value over the red value. The data was Agilent Lowess normalized: the values were corrected for the fact that RNA incorporates more green dye than red dye.

Period detection

To find periodicity in the data sets, the microarray time series for both the wild-type mice and knockout mice were run through the ARSER implementation. A period window of 20 hours to 28 hours was specified for the AR spectrum, as we found that this window produced the most interesting results.

ARSER can only use data where the samples are evenly-spaced: there must be a fixed time interval between each of the samples. The third day of the Bradfield dataset is not evenly spaced. To correct for this, points were linearly interpolated for the third day to create a constant 4-hour interval for the data. A disadvantage of linear interpolation is that it can tend to “smooth out” oscillating data, misrepresenting the peaks of the waveform and lowering the amplitude.

ARSER produces a q value, which represents the minimum false discovery rate at which the test is significant. For the clusterings in this experiment, we consider any gene with a q value of less than 0.01 to be cycling because it is important to have accurate estimates of the sinusoid parameters. When comparing distributions of parameters between the wild type and knockout sets, we consider genes with a q value less than 0.05 to be cycling, in order to increase our sample size.

Clustering sinusoids

Genes which were cycling were clustered using both k-means and agglomerative hierarchical clustering. The vectors used for clustering were the parameters of the sinusoids produced by ARSER (amplitude, period, phase.) A log scale was used to cluster amplitude because it has a heavily skewed distribution. Before clustering, each vector was “whitened.” Each field was scaled by its standard deviation to produce unit variance.

The data was clustered using both k-means and agglomerative hierarchical clustering algorithms with k values (the number of clusters) from 3 to 10. Because k-means clustering is not deterministic, it was run 20 times for each k value, and the clustering with the lowest average distance between its data points and center points was used.

Evaluation of sinusoid clusterings

Clusterings were evaluated using the method described by Ben-Hur et al., using CDistance described by Coen et al., as a distance metric between clusterings. Ten random subsamples of the wild type data (q <= 0.01) were generated, containing 75% of the original data points.

Each of the subsamples were clustered using both k-means and agglomerative hierarchical clustering for k = 3 to 10. The distances were measured between the clusterings of each subsample using CDistance, and the mean of all of the distances was taken, which acts as a measurement of the stability of each clustering.

Stability measurements were calculated separately for k-means clusterings and agglomerative hierarchical clusterings of the subsamples, and an additional set of stability measurements was calculated that included both k-means and agglomerative hierarchical clusterings.

Additionally, we queried the genes in each of our clusters against the Mouse Genome Informatics gene ontology database (Blake et al., 2011) to identify labels which have high frequencies within each cluster. Because the labels in the controlled vocabulary occur at different frequencies, we compare the frequencies of labels in our clusters to the frequency of labels in all annotated mouse genes.

Results

Cycle identification

The ARSER algorithm assessed the possible periodicity and fit a cosine wave to every gene in the 24,034-gene set. Table #1 shows examples of different cycling genes across a gradient of q values. The cycles appear to be less and less regular as q increases. Note that the q value only describes the probability of periodicity in the data, not the probability that the data follows a cosine wave.

For q = 0.05, the ARSER algorithm found 2,423 genes out of the total 24,034 that exhibited periodicity in the wild type mice. The mean period in the wild type was 24.44 hours (in the 20-28 hour period window,) with a standard deviation of 2.28 hours. The mean amplitude for the wild type was 0.11, with a standard deviation of 0.07. For phase, the mean was 11.14 hours and the standard deviation was 6.54 hours.

In the mice with the *Mop3* gene knocked out, there are far fewer genes with cycles identified by ARSER. There were only 41 genes found with q values less than or equal to 0.05. Of the 41 genes with cycles identified in the knockout, only 25 were cycling in the wild type.

With a maximum q value of 0.01, 568 of the 24,034 genes were cycling in the wild type. Only 7 were cycling in the *Mop3* knockouts. None of the 7 genes that were cycling in the knockouts were also cycling in the wild type.

Sinusoid parameter distributions

The two-sample Kolmogorov-Smirnov (KS) test was used to compare the distributions of the sinusoid parameters of cycling genes in the wild type and knockout mice. The KS test does not assume distributions for the two samples, and returns p values representing the probability that the two samples came from the same distribution.

For the periods identified in cycling genes ($p <= 0.01$) for the wild type (Fig. 4) and knockout (Fig. 5), the KS test returned a p value of 0.029, indicating that the distribution of periods is different between the two conditions.

A KS test returned a p value of 0.76 for the amplitudes in the wild type (Fig. 6) and knockout (Fig. 7) mice. For the phases in the wild type (Fig. 8) and knockout (Fig. 9) mice, a KS returned a p value of 0.47. The p values for the two parameters indicate that there is not a significant difference between the two distributions of each parameter in our data.

Sinusoid cluster evaluation and choice of k

Table 2 shows stability measurements for clusterings for k-means, agglomerative hierarchical, and both clustering algorithms for k values from 3 to 10, where k is the total number of clusters. The stability measurements are low, mostly below 0.5, indicating that that clusterings of our data tend to be stable, and indicating that there is clusterable order in our data.

The stability measurement is lowest (so stability is the highest) for the clustering generated by k-means at $k = 5$, so we choose 5 for our number of clusters.

Sinusoid clusterings

Figure #10 shows a dendrogram of an agglomerative hierarchical clustering of the sinusoid parameters for cycling genes ($q <= 0.01$) in the wild type mice. A “flattening” of the hierarchical clustering to produce 5 clusters produces a clustering similar to a k-means clustering with the same number of clusters.

The parameters of the sinusoids in the wild type mice can be clustered into 5 groups with a distortion of 0.9. The center points of the sinusoids are described in Table #3. The clusters are plotted in Figure #11.

A selection of notable terms from the gene ontology for each cluster is shown in Figure #12. The frequencies of annotations are different from the set of all annotated mouse genes and between clusters, which shows separation between the clusters.

Discussion

The number of genes that ARSER finds to be cycling decreases spectacularly between the wild type mice and the *Mop3* knockout mice. This change confirms the importance of *Mop3* in regulating circadian rhythms in mice.

ARSER identified 16 genes at q=0.05 and 7 genes at q=0.01 that were cycling in the *Mop3* knockout mice, but were not cycling in the wild type. We have not yet figured out why these genes are not cycling in the knockout, but it may be an anomaly of ARSER's q-value calculation. Further investigation of these genes may be interesting.

When comparing the distributions of sinusoid parameters between the wild type and knockout mice, there was only a significant difference between the distributions of the periods of the two conditions. The difference in the period is to be expected, because *Mop3* is responsible for many genes with a near-24-hour period. It is possible that the low number of genes in the knockout sample is responsible for the high p-values in for the other two parameters (amplitude and phase).

There is a “gap” in the distribution of periods in the wild type (Fig. 4) around 27 hours, where there are not any genes with a period falling in that interval. This is an anomaly of the way in which ARSER calculates periods. At the resolution of periods that ARSER considers, none of the possible periods fall within the interval. The gap is not visible in histogram of periods in the knockouts (Fig. 5) because the bins have wider intervals.

The clusterings of sinusoids of cycling genes in the wild type appear from the plots (Fig. 11) to be well-separated. The stability of the clustering indicate that the clusterings describe true order within the data. In the plot (Fig. 11) and in parameter distributions (Fig. 8), the phase appears to stratify the data the most of the parameters.

Based upon our evaluation of the clusterings, we can state with confidence that the genes in each cluster are regulated similarly. Identifying genes which are regulated similarly is an important step of mapping out regulatory networks of gene expression. Based upon our results, we can infer that some of the genes within our clusters are regulated together.

To further refine our clusterings, it may be helpful to implement other clustering algorithms which exploit different properties of the data to compare to our current clusterings. An effective ensemble clustering compares several different clustering algorithms to each other to find the optimal clustering.

One possible next step for uncovering the regulatory network behind circadian rhythms in gene expression is a physical network model analysis. A physical network model infers regulatory network structures from gene expression data under different conditions, and takes into account the effects of knocking out certain genes (Yeang et al., 2004). The model could be modified to better suit genes with periodic expression.

References

1. Akashi, M., and T. Takumi. 2005. The orphan nuclear receptor ROR alpha regulates circadian transcription of the mammalian core-clock Bmal1. Nat. Struct. Mol. Biol. 12:441-448. doi: 10.1038/nsmb925.
2. Ben-Hur, A., Elisseeff, A., and Guyon, I. A stability based method for discovering structure in clustered data. In Pacific Symp. on Biocomputing, 2002.
3. Blake JA, Bult CJ, Kadin JA, Richardson JE, Eppig JT and the Mouse Genome Database Group. 2011. The Mouse Genome Database (MGD): premier model organism resource for mammalian genomics and genetics. Nucleic Acids Res 39(suppl 1): D842-D848.
4. Bunger, M. K., L. D. Wilsbacher, S. M. Moran, C. Clendenin, L. A. Radcliffe, J. B. Hogenesch, M. C. Simon, J. S. Takahashi, and C. A. Bradfield. 2000. Mop3 is an essential component of the master circadian pacemaker in mammals. Cell. 103:1009-1017.
5. Coen, Michael H., M. Hidayath Ansari, and Nathanael Fillmore. "Comparing Clusterings in Space." Proceedings of the 27th International Conference on Machine Learning. Haifa, Israel. 2010. Print.
6. Darya Chudova, Alexander Ihler, Kevin K Lin. 2009. Bayesian detection of non-sinusoidal periodic patterns in circadian expression data. Bioinformatics 2009;25:3114-3120.
7. Smith, A. A., A. Vollrath, C. A. Bradfield, and M. Craven. 2008. Similarity Queries for Temporal Toxicogenomic Expression Profiles. PLoS Computational Biology. 4:ArteNo.:e1000116. doi: 10.1371/journal.pcbi.1000116 ER.
8. Smith, A. A., A. Vollrath, C. A. Bradfield, and M. Craven. 2009. Clustered alignments of gene-expression time series data. Bioinformatics (Oxford). 25:I119-I127. doi: 10.1093/bioinformatics/btp206 ER.
9. Straume, M. 2004. DNA Microarray time series analysis: automated statistical assessment of circadian rhythms in gene expression patterning, Methods Enzymol, 383, 149-166.
10. Yang, R., and Z. Su. 2010. Analyzing circadian expression data by harmonic regression based on autoregressive spectral estimation. Bioinformatics. 26:i168-i174. doi: 10.1093/bioinformatics/btq189.
11. C. Yeang, T. Ideker and T. Jaakkola. Physical Network Models. Journal of Computational Biology 11(2-3):243-262, 2004.

Tables and figures

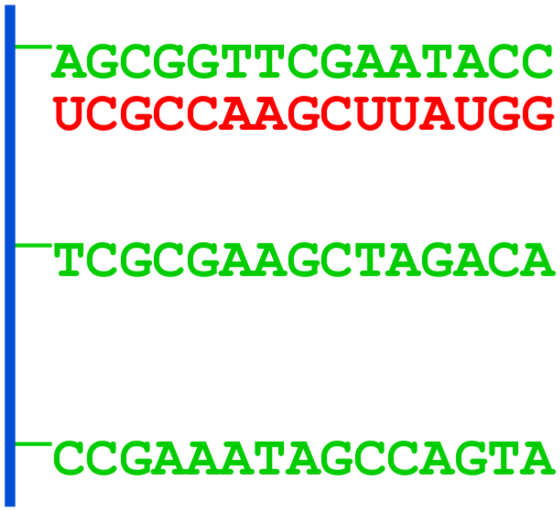


Figure 1: *Depiction of a microarray, showing RNA sequences embedded on a chip. (Source: Colin Dewey, lecture notes for Introduction to Bioinformatics – Fall 2010)*

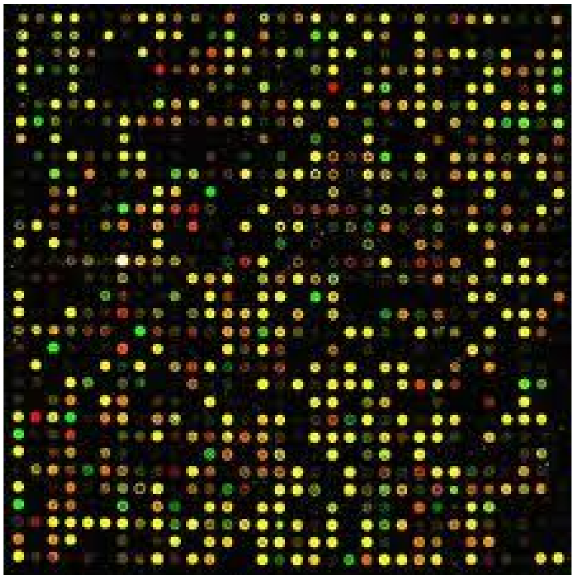


Figure 2: *Image of a microarray showing florescent tags. (Source:Colin Dewey, lecture notes for Introduction to Bioinformatics – Fall 2010)*

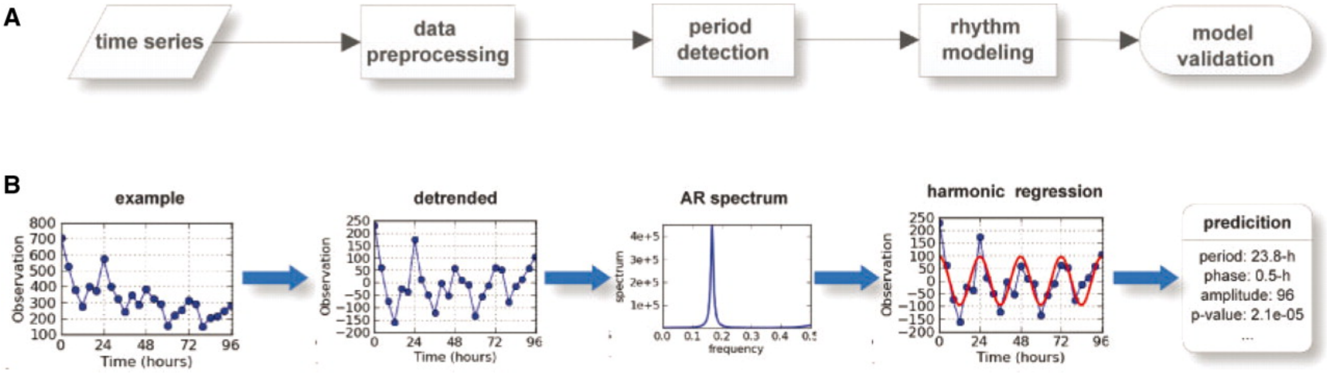
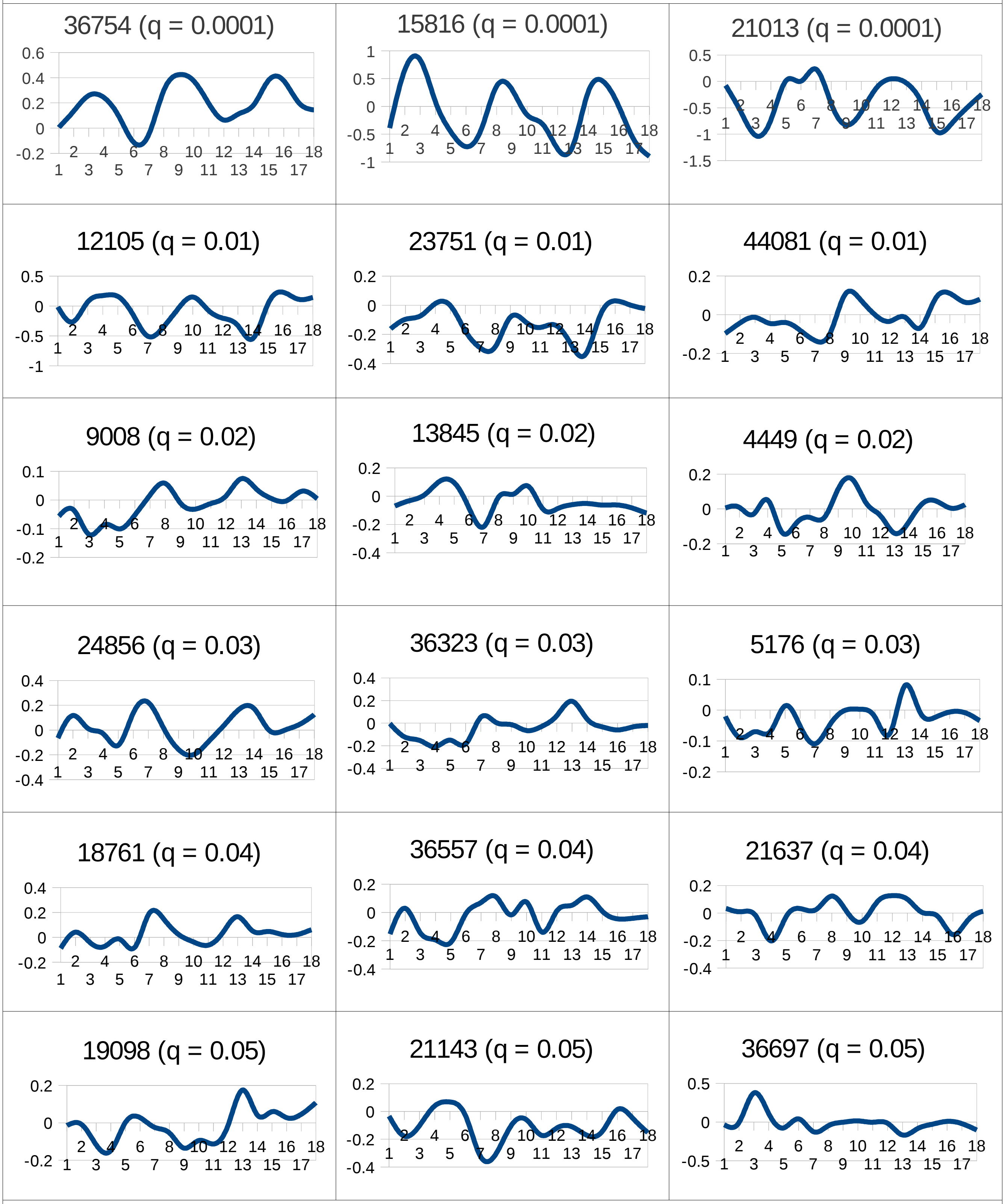


Figure 3: *Diagram of ARSER process. First the data is detrended and smoothed, then fitted with an AR spectral model to find the period. A harmonic regression is computed, and sinusoid parameters are returned. (Source: Yang and Su, 2010)*

Table 1: *Examples of cycling genes with different ARSER q values*



Each figure depicts the expression levels of one gene over time. Numbers on the X axis are sample numbers (4 hours apart for 3 days.) ARSER q values and figures use interpolated data for 3rd day.

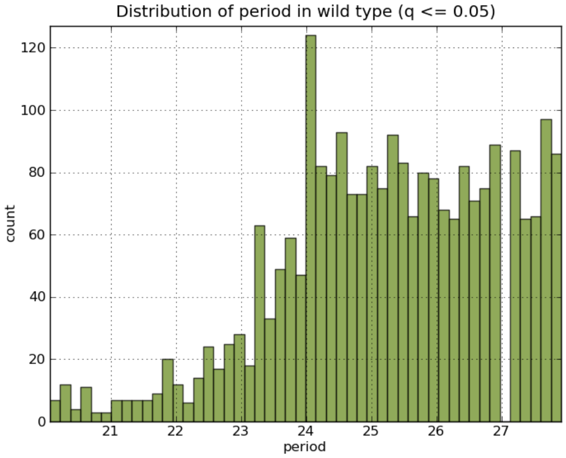


Figure 4: *Histogram of periods (in hours) of cycling genes (q <= 0.05) in wild type mice*

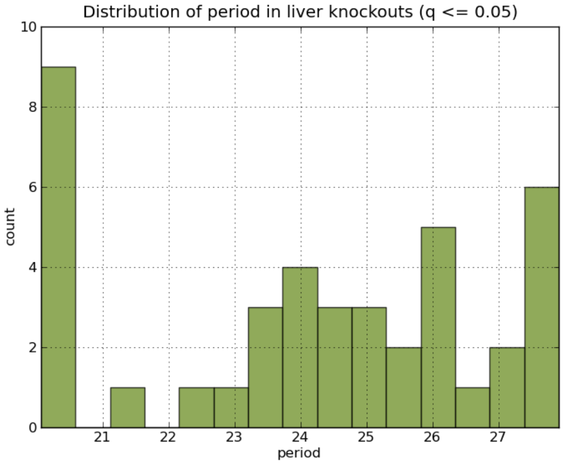


Figure 5: *Histogram of periods (in hours) of cycling genes (q <= 0.05) in knockout mice*

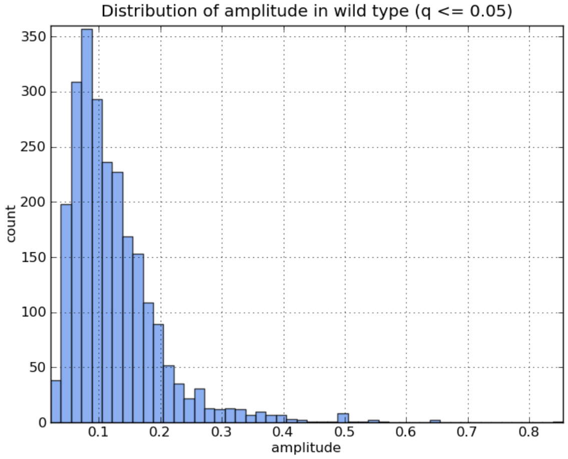


Figure 6: *Histogram of amplitudes of cycling genes (q <= 0.05) in wild type mice*

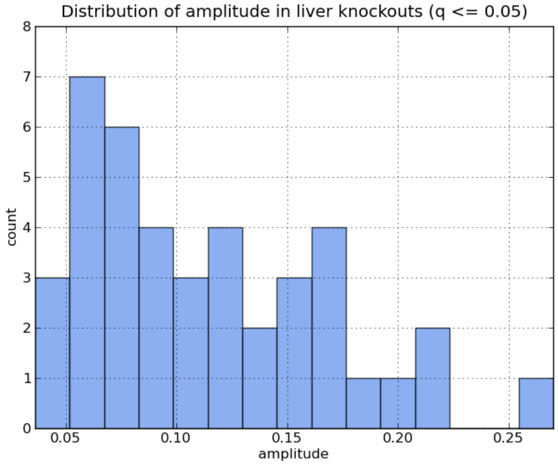


Figure 7: *Histogram of amplitudes of cycling genes (q <= 0.05) in knockout mice*

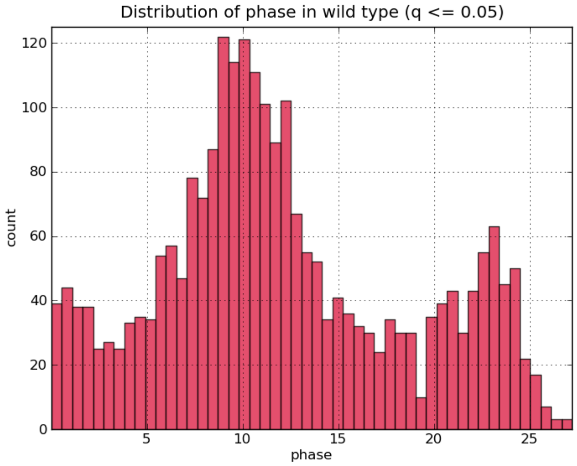


Figure 8: *Histogram of phases (in hours) of cycling genes (q <= 0.05) in wild type mice*

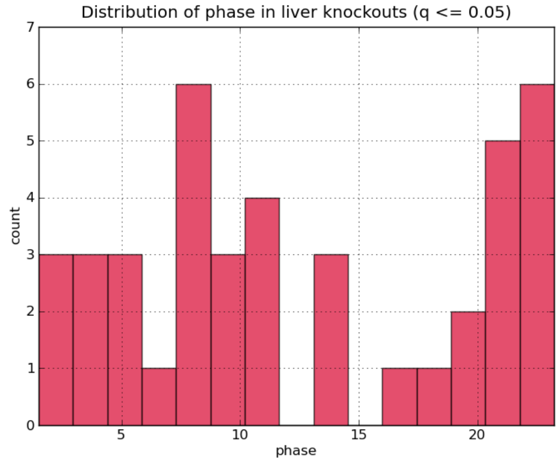


Figure 9: *Histogram of phases (in hours) of cycling genes (q <= 0.05) in knockout mice*

k	kmeans	hierarchical	both
3	0.38	0.85	0.6566
4	0.4	0.6	0.537
5	0.31	0.47	0.448
6	0.33	0.48	0.4585
7	0.38	0.48	0.4619
8	0.38	0.45	0.4466
9	0.35	0.42	0.4262
10	0.34	0.42	0.4237

Table 2: *Stability measurements by clustering algorithm and number of clusters (k) for wild type sinusoid clustering. Stability was calculated by clustering random subsamples of the full dataset, and then averaging the distances between all of the clusterings. A low measurement indicates that the clustering is stable.*

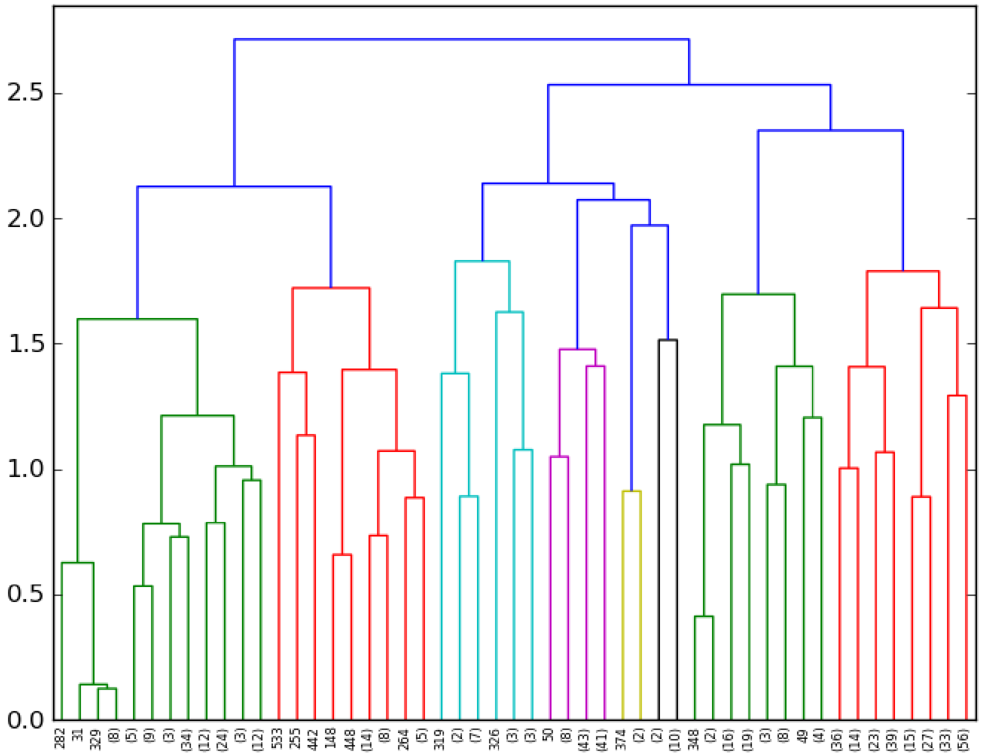


Figure 10: *Dendrogram of a hierarchical clustering of sinusiouds for cycling genes in wild-type mice. X-labels in parentheses indicate the number of genes in that cluster. X-labels without parentheses indicate a single gene. The Y axis represents euclidean distance between the clusters.*

Cluster ID	Period (hours)	Amplitude (log)	Phase (hours)
1	19.92	-4.01	1.16166847
2	18.73	-2.4	1.38560658
3	18.39	-2.33	2.97635291
4	17.71	-3.74	0.60074863
5	18.78	-4.09	2.85

Table 3*: Codebook for k-means clustering of sinusoids of cycling genes (q <= 0.01) in wild type mice. The values in each cell represent the values of each parameter for the center point of each cluster. Amplitude is clustered on a log scale. Values are normalized for unit variance.*

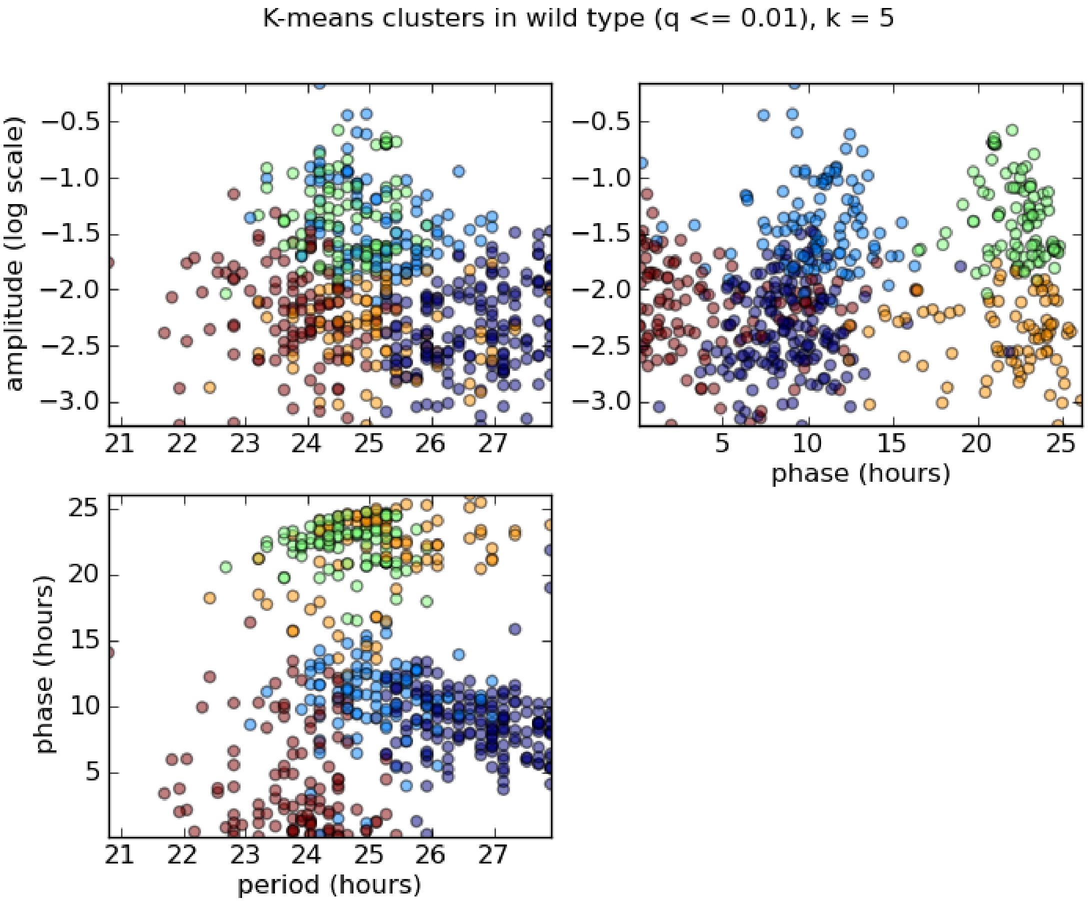
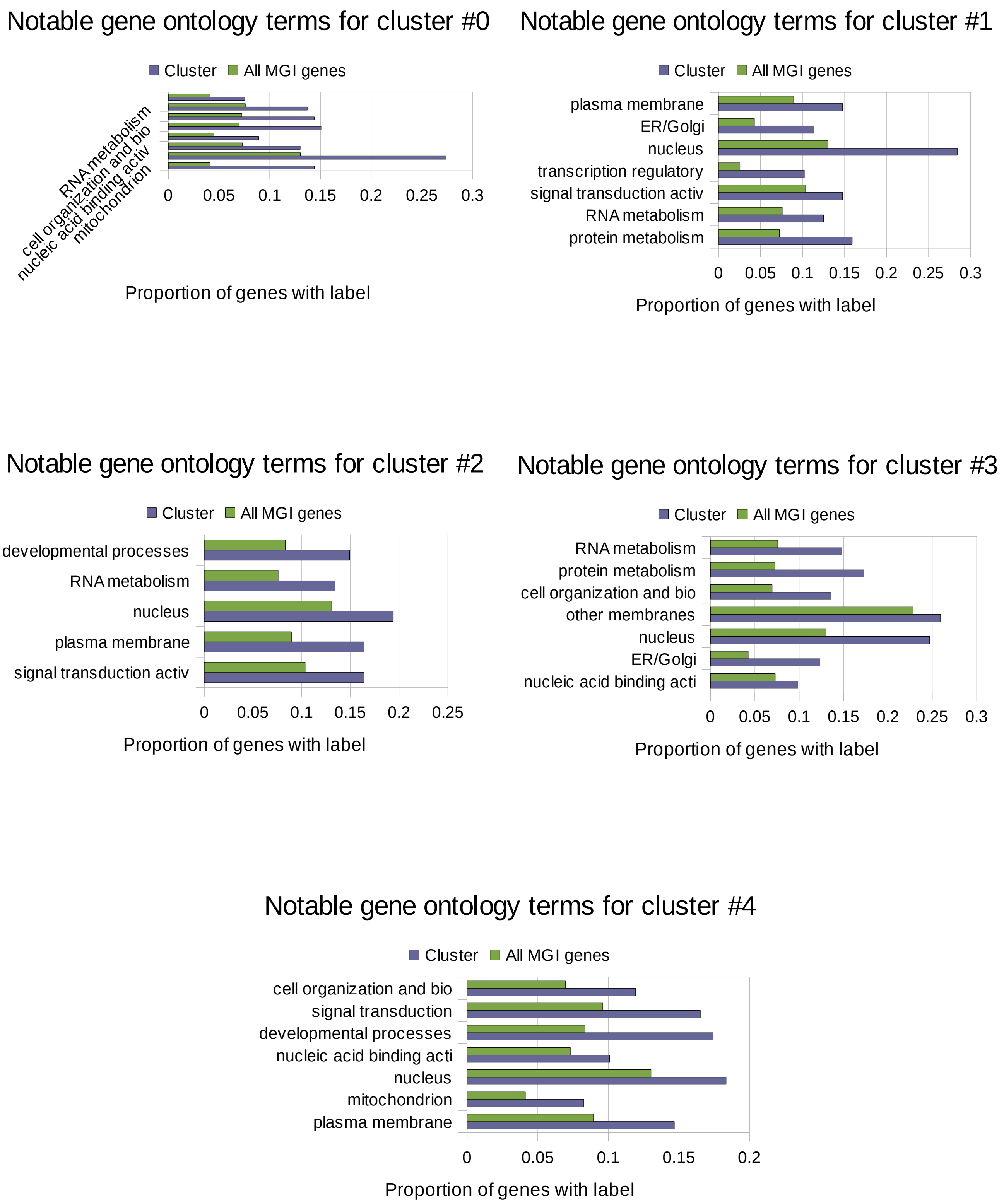


Figure 11*: Plot of clustering of sinusiods for cycling genes (q <= 0.01) in wild type mice. Colors represent cluster assignment.*

Figure 12: *Notable gene ontology terms for genes in sinusoid clusters. A sample of terms are selected so that the frequency of the term within the cluster is different from the frequency of the term within all annotated mouse genes.*


Notable gene ontology terms for cluster #0
Cluster
All MGI genes
RNA metabolism
cell organization and bio
nucleic acid binding activ
mitochondrion
Proportion of genes with label
Notable gene ontology terms for cluster #1
Cluster
All MGI genes
plasma membrane
ER/Golgi
nucleus
transcription regulatory
signal transduction activ
RNA metabolism
protein metabolism
Proportion of genes with label
Notable gene ontology terms for cluster #2
Cluster
All MGI genes
developmental processes
RNA metabolism
nucleus
plasma membrane
signal transduction activ
Proportion of genes with label
Notable gene ontology terms for cluster #3
Cluster
All MGI genes
RNA metabolism
protein metabolism
cell organization and bio
other membranes
nucleus
ER/Golgi
nucleic acid binding acti
Proportion of genes with label
Notable gene ontology terms for cluster #4
Cluster
All MGI genes
cell organization and bio
signal transduction
developmental processes
nucleic acid binding acti
nucleus
mitochondrion
plasma membrane
Proportion of genes with label