SUPPLEMENTAL MATERIAL

Glossary of machine learning terms

Accuracy measures the ratio of correct predictions, both positive and negative cases, to the total number of cases evaluated.

Associative memories are content-addressable memories constructed by observing co-occurrences of data that are capable of retrieving a piece of data upon presentation of only partial information from *that* piece of data (auto-association), or recall an associated piece of datum from *one* category upon presentation of data from *another* category (hetero-association).

Classifier provides a mapping from unlabeled instances characterized by a set of variables to a category label.

Data binning is a data pre-processing technique places a continuous value within a given interval, and replaces the original value with a value representative of that interval.

Data normalization refers to the creation of shifted and scaled versions of statistics, where the intention is that these normalized values allow the comparison of corresponding normalized values for different datasets.

K-Nearest Neighbor **classifier** classifies an object by a majority vote of the fixed number (k) of its nearest neighbors, with the object being assigned to the class most common among its voting neighbors.

Learning curve refers to a plot of the *prediction accuracy/error* of a machine-learning algorithm vs. the *training set sizes used.*

Random forests classifier is an ensemble learning method for classification. It constructs a multitude of decision trees at training time and outputting the class that is the mode of the classes.

Support Vector Machine is a discriminative classifier outputs an optimal hyper-plane from training data, which categorizes new examples.

Wrapper method takes advantages of the prediction performance of a given machine-learning algorithm to assess the relative usefulness of different subsets of variables.

Supplemental Methods

Structure of a speckle tracking echocardiographic (STE) measurement

The STE software measures each parameter at multiple spatial locations within the myocardium, and at multiple time-points within the cardiac cycle. Figure S-1 depicts three-dimensional structure of a typical strain measurement. The x-axis is the elapsed time in milliseconds and y-axis is the spatial locations within the myocardium. Each spatial location is identified with a prefix 's' followed by a location number. The z-axis is the strain measurement in percentage.

Data Discretization

The associative memory classifier (AMC) accepts discrete data only. To use the classifier, all STE data were binned using quintiles. The thresholds for the quintile were derived from a comparison cohort of 47 control subjects with no structural heart disease. For clinical and conventional echocardiographic data, the numerical values were discretized by quintiles based on the distribution of the available data.

Each parameter could be discretized by different number of bins. Our experiments have shown that the choice of number of bins has limited effect on the average accuracies. Therefore, for simplicity, all the parameters were discretized with the same number of bins, and the optimal number of bins was determined by testing classification accuracy using the AMC. As depicted in Figure S-2, the accuracies vary slightly over different number of bins used for discretization, with quintile binning being the best, albeit marginally only. However, the relative accuracy for different binning approaches remains nearly same at different training fractions.

Variable selection

Methods of classification often rely on computing and comparing distances between objects from different classes with overall similar properties. With more than 1,800 STE data points, each patient is easily dissimilar from each other on many counts, resulting in a large variance in classification accuracy. With a relatively small sample of patients and a large degree of freedom in the variables, challenges are created in data modeling and analysis using statistical significance, which is often known as the "curse of dimensionality".[1] One approach to alleviate this problem is through variable selection. Variable selection helps achieve better predictive performance with reduction of computing resources and time. Although the variable selection scheme described in this paper is in the context of differentiating constrictive pericarditis (CP) and restrictive cardiomyopathy (RCM), this methodology is applicable to other heart diseases as well.

We devised a multi-level variable selection scheme based on the wrapper method.[2] This methodology uses a machine learning approach to assess the usefulness of subsets of variables. Although several different search strategies have been proposed in the past[3], we elected a general strategy of greedy forward selection based on a variable-ranking criterion. The scheme includes best-ranked variables into nested growing subsets until accuracy assessed from the wrapper ceases to improve the selection.[4] We used AMC as the wrapper and the L^1-distance as the ranking criterion.

The L^1-distance based greedy forward selection ranking criterion performed well for clinical and conventional echocardiographic data. Figure S-3 depicts the ROC curves for assessing the diagnostic performances of the top two (B2), three (B3), and four (B4) clinical and echocardiographic variables. The optimum selection was reached once the fourth variable was included as there was no further improvement in accuracy with the addition of more variables.

In contrast, the STE data exhibited a non-linear behavior with fluctuations in the accuracy in the direction of selection. To overcome this and to allow for optimum STE variable selection, we used a step-wise approach (Figure S-4). First, the cardiac cycle time intervals having the highest accuracy were identified, which were then correlated with each other (Figure S-5). The three subsets at cardiac cycle percentages 15% (t_3), 20% (t_4), and 60% (t_{12}) demonstrated the highest accuracy (Figure S-4A) but the variable subset t_{12} was essentially uncorrelated with any variable in subsets t_3 or t_4 (Figure S-

5). However, when these subsets were combined, all the different combinations yielded better performance than any individual subset with the highest accuracy achieved by combining $t_{3/4}$ (one value from either t_3 or t_4 for each data point) with t_{12} (Figure S-4B). Because of fluctuation in the accuracy assessment, this combined variable subset "$t_{3/4} + t_{12}$" was further refined by selecting variables only in the regions where addition of more variables resulted in an incremental gain in accuracy. Eighty STE variables were thus selected (Figure S-4C).

However, when these 80 STE variables were added to the top four clinical and conventional echocardiographic variables, paradoxical reduction in accuracy was observed. The reduction in accuracy occurred because of redundancies and unwanted interactions in the combined subset. To remove unfavorable variables in the combined subset, we merged the ranking of STE and clinical/ conventional echocardiographic variables, and used the least favorable clinical variable (heart rate) as a starting point (probe) to exclude the lower ranked STE variables one-by-one until the diagnostic performance stops improving, reducing STE variables from 80 to the final 15.

L^1-Distance

The quality of variable selection greatly depends on its ranking criteria. Various variable-ranking criteria have been proposed in the past, such as mutual information, correlation criteria, and Fisher's criterion.[4] We used a criterion based on *L^1-distance,* which estimates the classification rate of a single variable. This criterion is captured using the difference between the probability distribution (PD) of two class labels. For example, the difference between PDs can be expressed in *Kullback-Leibler divergence*.[5] Other divergence definitions have been proposed in practice to achieve better computation efficiency, numerical stability, and robustness against outliers.[5] For our purposes, we define *L^1-distance* between two PDs as the following-

$$L^1(P,Q) = \int |P(x) - Q(x)|\,dx \approx \sum_{x_i} |P(x_i) - Q(x_i)| \cdot \Delta x$$

P(x) and *Q(x)* are two probability distributions for the same variable *x*. By definition, the area under a PD is one, therefore, the value of *L^1-distance* is bounded between zero and two [$0 \le L^1(P,Q) \le 2$]

(Figure S-6). One could multiply the constant ½ to scale the *L^1-distance* value to a value between zero and one for estimating the probability of a variable that can discriminate between two classes across all its values. It also approximates *classification rate* (accuracy) of a single variable, which is the sum of true positives and true negatives of prediction divided by the total number of trials. In a binary classification (e.g. CP/RCM), the overall accuracy is identical to the accuracy of predicting either event (CP or RCM) because of symmetry. The larger distance is indicative of a valuable variable. We used *L^1-distance* as the ranking criterion to sort all the variables creating the default order of inclusion for variables.

Training of AMC and assessment of its predictive accuracy

The concept of AMC originated from Hopfield network and sparse distributed memory.[6, 7] We used the Natural Intelligence Platform (NIP) offered by Saffron Technology, Inc., providing the cognitive computing classification algorithm using auto-associative memory to learn the training data and provide prediction.[8]

The input to an associative memory classifier is a set of predictors. An *attribute* is a pair of name and value, denoted as *name:value*, representing a variable and its value, for examples, *age:24* and *gender:male*.

AMC Training

We use x_i to represent an attribute and define S_{all} as the set containing all possible N_A predictors in a problem space. Therefore, any predictor set S_{input} is just a subset of $S_{all\text{-}}$

$$S_{all} = \{x_i \mid i = 1, \ldots, N_A\}$$
$$S_{input} \subseteq S_{all}$$

During the training phase, a class label is accompanied with the predictor set. The associative memory of the class label will *observe* all pairwise associations between the predictors. To help explain the observation of associations, we define an predictor vector V of the size N_A such that each element in the vector corresponds to an attribute in S_{all}. An element in the vector is set to one if the corresponding attribute is in the predictor set; otherwise, the element is set to zero.

$$V = (v_1, v_2, ..., v_k, ..., v_N)$$
$$v_k = \begin{cases} 1, x_k \in S_{input} \\ 0, x_k \notin S_{input} \end{cases}$$

All the pairwise associations of predictors can be represented as a $N_A \times N_A$ matrix *A*. Element a_{ij} in the matrix is set to one if both v_i and v_j are set to one in the predictor vector. The association matrix *A* can be created by multiplying the transpose of predictor vector and the predictor vector. Since predictor vector has only elements 0 and 1, association matrix *A* is a zero-one matrix, with the elements having value either 0 or 1. Following is the formal definition of the association matrix *A*-

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} = V^T V$$
$$a_{ij} = v_i * v_j,\ i, j = 1..\ N_A$$

The result of observing all N_{c_m} predictor vectors for a class label c_m is the sum of all its predictor association matrices A_k. The resultant matrix, denoted as M_{C_m}, is what we referred as the associative memory for class label c_m. It is an integer matrix with the dimension $N_A \times N_A$.

$$M_{c_m} = \sum_{k=1}^{N_{c_m}} A_k = \sum_{k=1}^{N_{c_m}} V_k^T V_k$$

After observing all the training vectors, we would have a set of associative memories, one for each class label. For a set of N_{CL} class labels (S_{CL}), the set of all the associative memories S_{AM} is the following-

$$S_{CL} = \{c_m \mid m = 1 \ldots N_{CL}\}$$
$$S_{AM} = \{M_{c_1}, M_{c_2}, \ldots, M_{c_{N_{CL}}}\}$$

AMC Prediction

The predictor vector is scored against the associative memory of each class label. The class label with the best score is selected as the prediction. We used the default option offered in Saffron NIP to compute the score. In the default option, two scores are calculated when evaluating a predictor vector against an associative memory, the cardinality (N_{ma}) and the mutual information of matched associations. The class labels are ranked by the cardinality followed by the mutual information scores. The heuristics is that the more matched associations, the better the predictor vector belongs to a class label.

We can express matched associations between an predictor vector V_k and associative memory of a class label as a matrix M_{A_k,c_m} by applying *Hadamard Product*[1] between its predictor association matrix A_k and the class associative memory M_{C_m}. The associative memory M_{C_m} behaves like a Boolean mask to filter associations in A_k.

$$M_{A_k,c_m} = A_k \circ M_{c_m}$$

We define a function *B(x)* by converting a numerical value *x* into one if *x* is a non-zero value. It can be applied to each element of a matrix.

$$B(M) = [B(m_{ij})]$$
$$B(x) = \begin{cases} 1, & x \neq 0 \\ 0, & x = 0 \end{cases}$$

[1] Let X and Y be $m \times n$ matrices. The *Hadamard Product* of X and Y is defined by $[X \circ Y] = [x_{ij} * y_{ij}]$, for all $1 \leq i \leq m$, $1 \leq j \leq n$.

Therefore, we can represent the cardinality (N_{ma}) of matched association of an predictor vector to a class associative memory as the following-

$$N_{ma} = \sum_{\substack{\forall m_{ij} \in M_{A_k,C_m} \\ 1 \le i < N,\ i < j \le N}} B(m_{ij})$$

Since matched association matrix is symmetrical, the cardinality of matched associations is the number of the non-zero elements in the upper triangular matrix above the diagonal. The associations at the diagonal (m_{ii}) can be interpreted as matched predictors in the class associative memory. For simplicity and to value matched associations over the predictors, Saffron NIP cardinality score (SC_{c_m}) of the predictor vector against a class label c_m is the total count of non-zero elements in the matched association matrix-

$$SC_{c_m} = \sum_{\forall m_{ij} \in M_{A_k,C_m}} B(m_{ij})$$

In the event, there are ties at the cardinality scores; Saffron NIP uses mutual information of matched associations to break the ties. The following formula defines the mutual information score of the matched associations (MI_{c_m}), which can be calculated from M_{A_k,c_m} with log(0) or log(0/0) treated as zero-

$$MI_{c_m} = I(X,Y \mid C = c_m) = p(c_m) \sum_{y \in S_{input}} \sum_{x \in S_{input}} p(x,y \mid c_m) \log \frac{p(x,y \mid c_m)}{p(x \mid c_m) p(y \mid c_m)}$$

For ease of comparison, Saffron AMC uses a trick concatenating above two metrics into one binary bit vector by shifting SC_{c_m} into the high-order bits while keeping MI_{c_m} at the low-order bits, which can be interpreted as a single classification score (R_{cm})-

$$R_{cm} = (SC_{cm} << \ \omega) + MI_{cm} = SC_{cm} * 2^{\omega} + MI_{cm}$$

The symbol << represents the shift-bits-right operator shifting binary representation of a value toward high-order bits by ω bits, which is equivalent to multiplying by 2^{ω}. The trick works as long as the maximum of all MI_{c_m} is less than 2^{ω} in the problem space.

For binary classification, we can define a normalized score Y_{cl} with the classification scores of CP and RCM to predict the class label as the following-

$$Y_{cl} = \frac{R_{cp} - R_{rcm}}{\max(R_{cp}, R_{rcm})}$$
$$Y_{cl} > 0 \rightarrow CP, Y_{cl} < 0 \rightarrow RCM$$

Assessment of AMC accuracy

Because of the small number of trials available in the data, we assessed variable subsets by running cross validation tests over many randomized training/test splits for stable results and to spot overfitting. Total trials were randomly divided into 10 partitions, which were re-partitioned at different rounds of tests. The training and test data were assembled from the partitions according to the desired ratio between the training and test data. For training fraction of 0·7, which is 70% and 30% split between the training and test data, randomly selected seven partitions were combined into the training set while the remaining three partitions were assembled as the test set. We assessed performances using non-repeating bootstrap samples and cross validation tests over many rounds to form distribution for computing errors and confidence intervals. Accuracies reported from the bootstrap samples and cross validation tests were cross-referenced to ensure consistency of the results. By averaging 10 tests over non-repeating bootstrap samples, the run yielded result equivalent to 10-fold cross validation at the training fraction of 0·9.

K-fold cross validation tests

K-fold cross validation scheme divided the data into non-overlapped K partitions. Each partition was rotated to be the test set and the rests are used as training data. The accuracy was calculated by averaging the accuracies over *K* tests. To reduce variability, multiple rounds of cross-validation were performed and averaged. The 2-fold and 10-fold cross-validation tests were corresponding to the training fractions of 0·5 and 0·9 respectively. To perform cross validation for the fractions that did not fall at exact *1/K*, the number of tests was calculated by rounding the number of 1/(1-fraction). For example, for the training fraction 0·7, the accuracy would be averaged over three test runs (1/0·3 ≈ 3) with non-repeating samples.

Holdout validation

We randomly hold out 50% of the patients from each class to reduce the total number of patients to 44 with 22 CP and 22 RCM patients respectively. With the reduced data set, we performed variable ranking using L1 distances and selected top 15 STE variables or top 4 ECHO clinical variables depending on the configuration of the run. In the manuscript, we used the wrapper method to further optimize the selection of the variables. To be conservative on our results, we avoided the step using wrapper method. If single variable is required for the run configuration, such as mitral annular e' velocity, we introduced the variable directly. The final selected variables were used for all learning algorithms to predict the class labels for the 50% holdout patients.

SUPPLEMENTARY FIGURE LEGENDS

Figure S-1: Three-dimensional structure of a typical strain measurement. s_N refers to a specific spatial segment location within the myocardium.

Figure S-2: The effect of number of data bins on the predictive accuracy of a variable

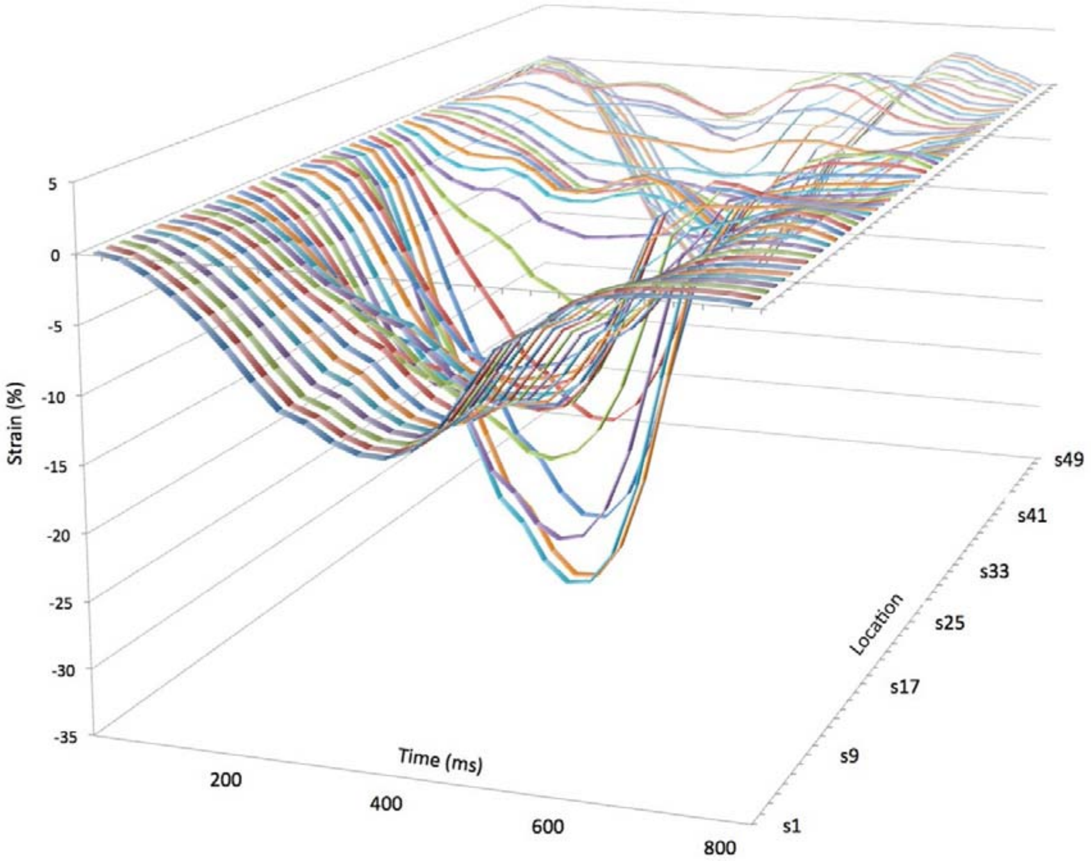
Figure S-3: The receiver operating characteristics curves for assessing the differences in diagnostic performances of the top two (B2), three (B3), and four (B4) clinical and conventional echocardiographic variables. (A) B3 resulted in statistically significant improvement in diagnostic performance as compared to B2; (B) further improvement in diagnostic performance with B4 was less marked, but was still statistically significant.

Figure S-4: The step-wise approach to selection of speckle tracking echocardiography (STE) variables. (A) STE variables were selected from cardiac cycle phases that provided maximum accuracy. Thus, a total of 270 STE variables from three best cardiac cycle phases (t_3 , t_4 and t_{12}) were selected; (B) accuracies of different subset combinations from above were assessed and redundant variables were removed, leaving a total of 180 STE variables; (C) Further selection of the STE variables that contributed to the ascending accuracy in the curve by the order of L^1-distance ranking. A total of 80 STE variables were thus selected. Please see text for more details.

Figure S-5: Correlation between speckle tracking echocardiographic variables derived from three different cardiac cycle time intervals (t_3, t_4, and t_{12}). As is evident, strong correlation was seen between t_3 and t_4 subsets (left side panels) but not between t_3 and t_{12} subsets (right side panels). Seg-N refers to a specific spatial segment location within the myocardium and t_N refers to a specific time interval within the cardiac cycle.

Figure S-6: The concept of L^1-distance ranking. L^1-distance estimates the discriminatory ability of a single variable by measuring the non-overlapping area between the two resultant probability distributions for the two outcomes [in this example, constrictive pericarditis (CP) and restrictive cardiomyopathy (RCM)]

Figure S-1



5
0
-5
-10
-15
-20
-25
-30
-35
Strain (%)
200
400
600
800
Time (ms)
s1
s9
s17
s25
s33
s41
s49
Location

Figure S-2

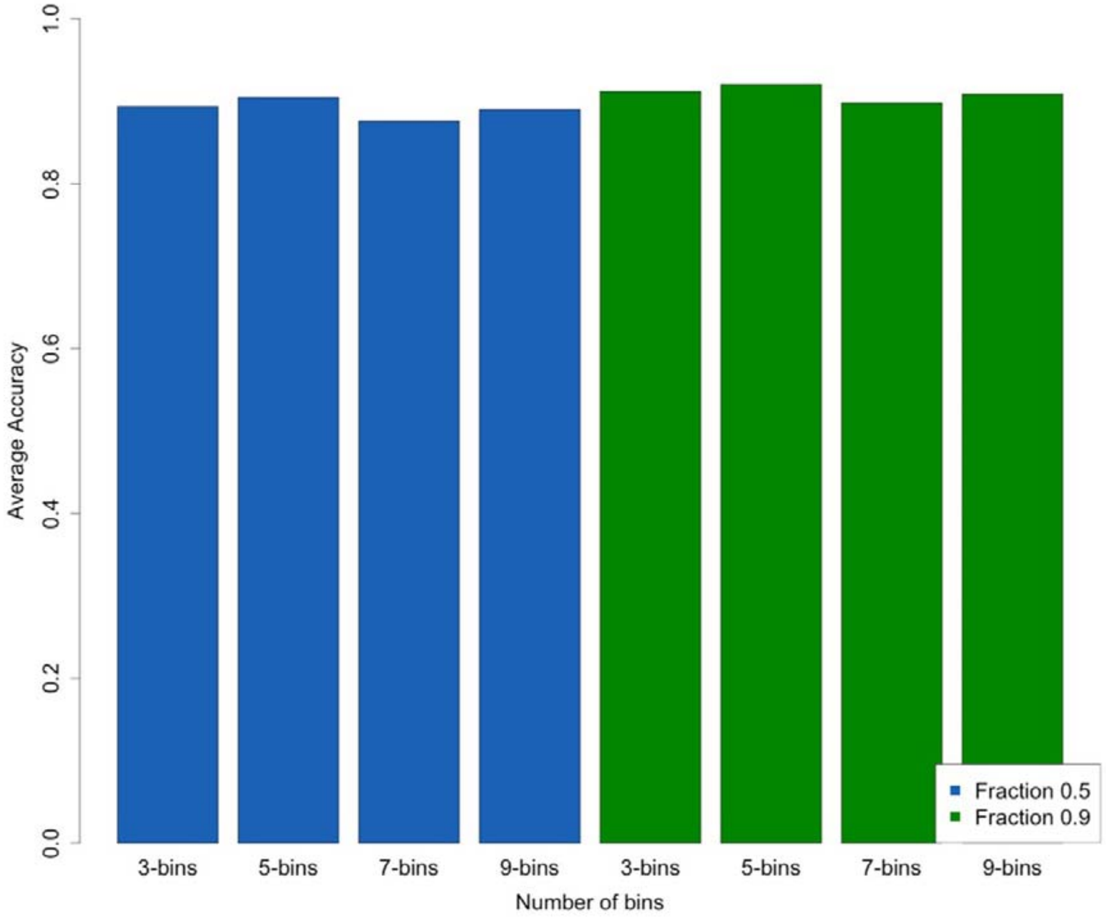

1.0
0.8
0.6
0.4
0.2
0.0
Average Accuracy
3-bins
5-bins
7-bins
9-bins
3-bins
5-bins
7-bins
9-bins
Number of bins
■ Fraction 0.5
■ Fraction 0.9

Figure S-3

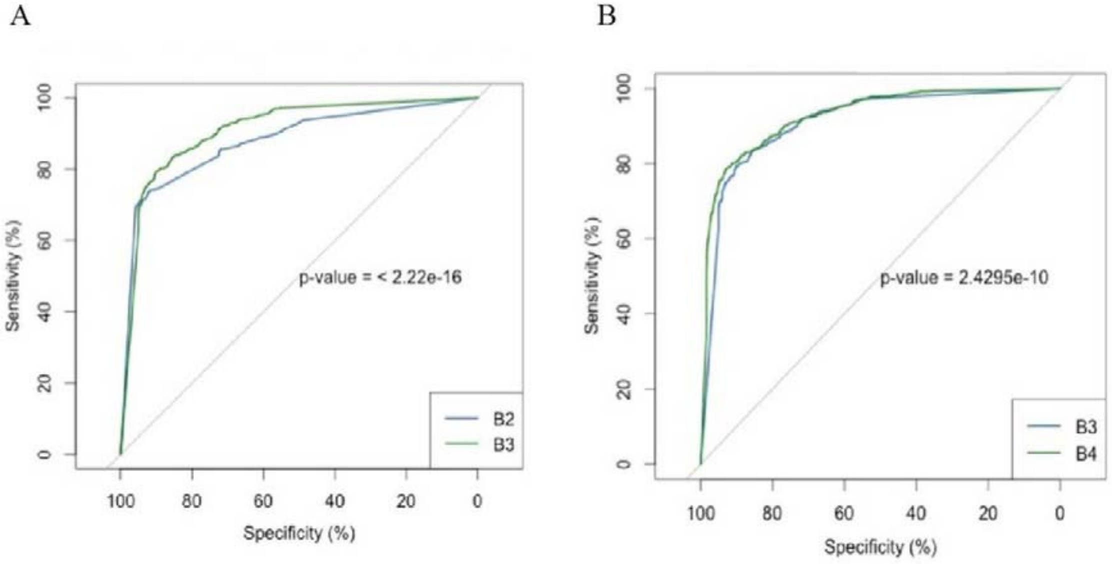

A
B
Sensitivity (%)
Specificity (%)
0
20
40
60
80
100
p-value = < 2.22e-16
p-value = 2.4295e-10
B2
B3
B3
B4

Figure S-4

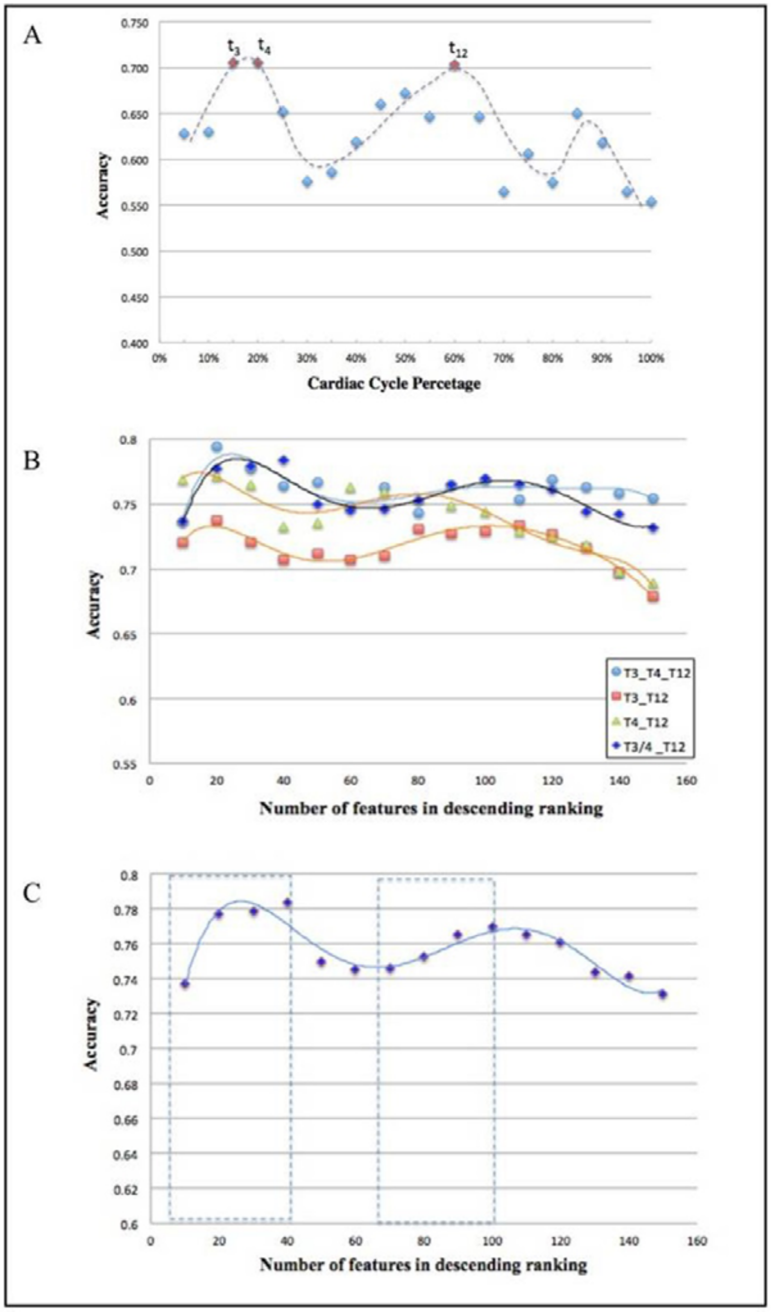
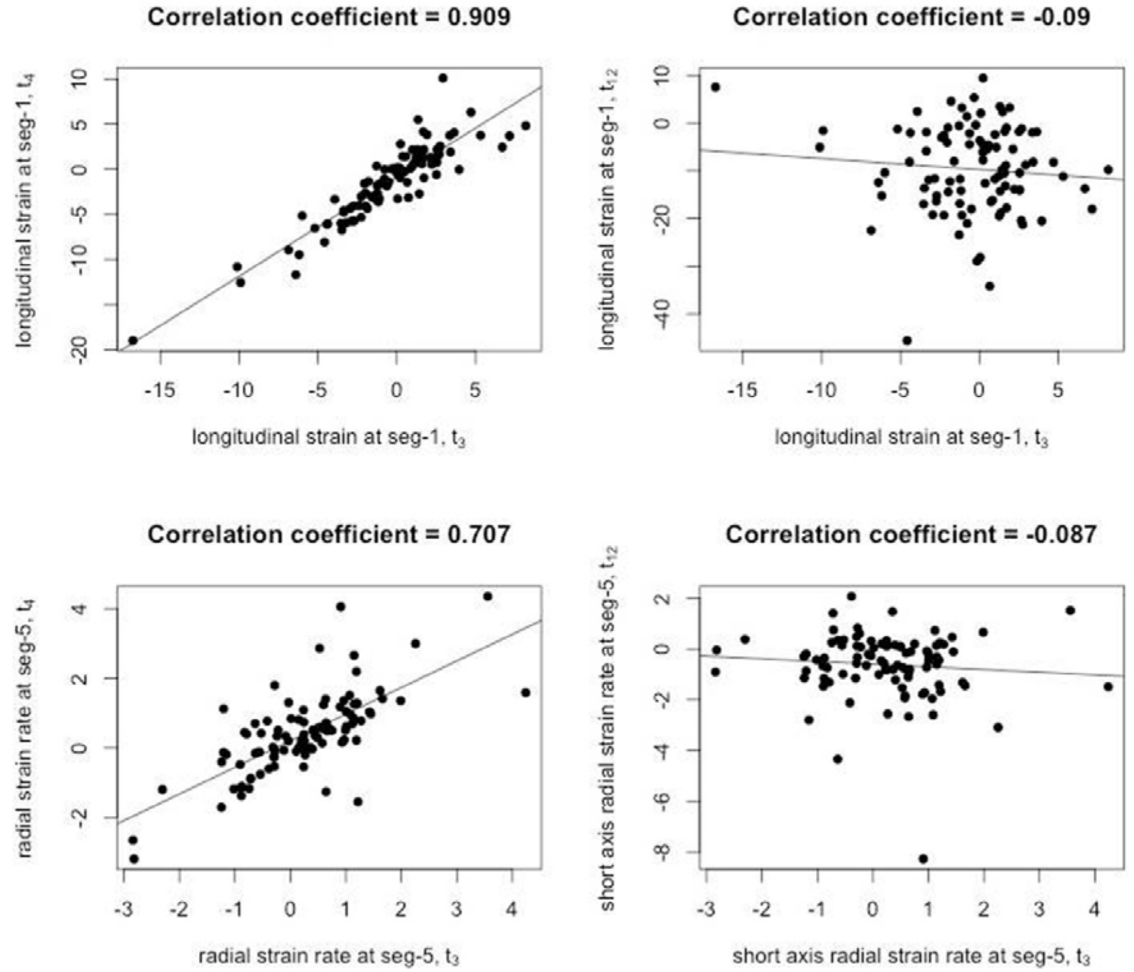

A
0.750
0.700
0.650
0.600
0.550
0.500
0.450
0.400
t_3
t_4
t_{12}
Accuracy
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Cardiac Cycle Percetage
B
0.8
0.75
0.7
0.65
0.6
0.55
Accuracy
● T3_T4_T12
■ T3_T12
▲ T4_T12
◆ T3/4 _T12
0
20
40
60
80
100
120
140
160
Number of features in descending ranking
C
0.8
0.78
0.76
0.74
0.72
0.7
0.68
0.66
0.64
0.62
0.6
Accuracy
0
20
40
60
80
100
120
140
160
Number of features in descending ranking

Figure S-5



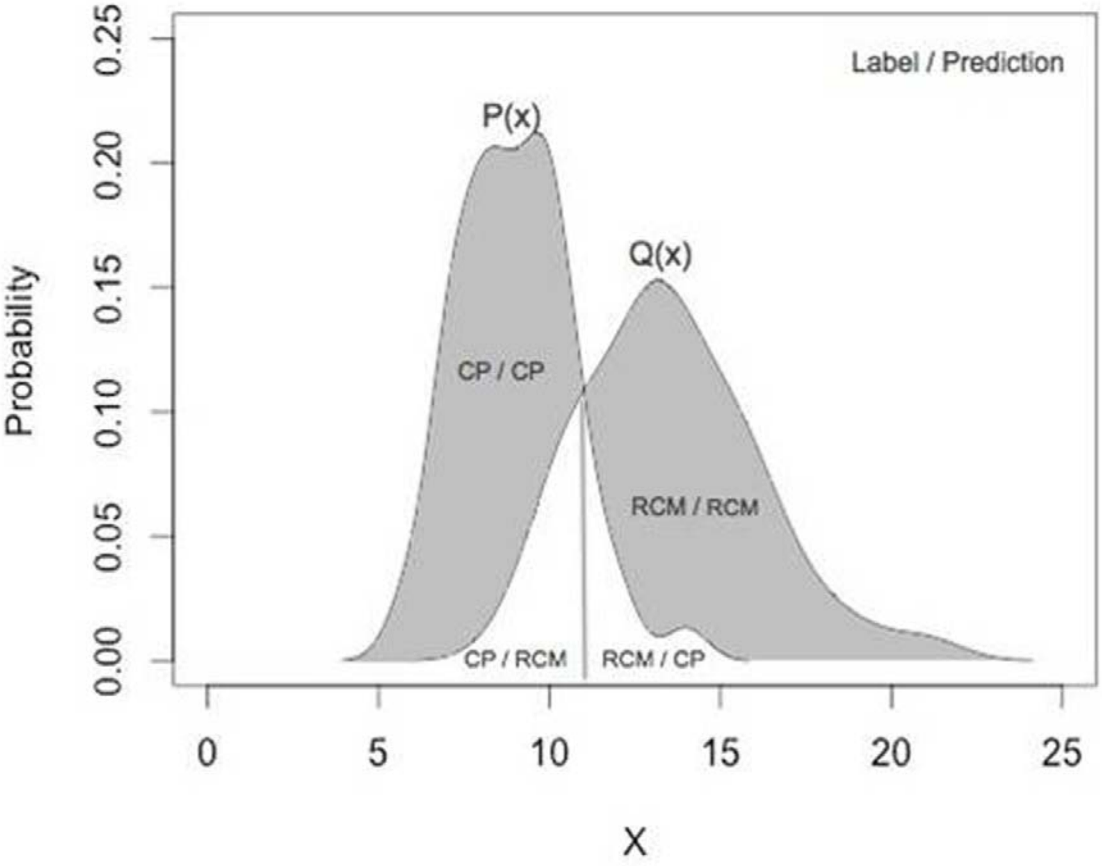
Correlation coefficient = 0.909

Correlation coefficient = -0.09

Correlation coefficient = 0.707

Correlation coefficient = -0.087

Figure S-6


Label / Prediction
P(x)
Q(x)
CP / CP
RCM / RCM
CP / RCM
RCM / CP
Probability
0.00
0.05
0.10
0.15
0.20
0.25
0
5
10
15
20
25
X

SUPPLEMENTAL REFERENCES

1. Bellman RE. Dynamic Programming. New Jersey: *Princeton University Press*; 1957.
2. Kohavi R, John GH. Wrappers for feature subset selection. *Artificial Intelligence*. 1997; **97**: 273–324.
3. Blum AL, Langley P. Selection of relevant features and examples in machine learning. *Artificial Intelligence*. 1997; **97**: 245–71.
4. Guyon I, Elisseeff A. An introduction to variable and feature selection. *The Journal of Machine Learning Research*. 2003; **3**: 1157–82.
5. Kullback S, Leibler RA. On information and sufficiency. *The Annals of Mathematical Statistics*. 1951; **22**: 79-86.
6. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci U S A*. 1982; **79**: 2554–8.
7. Snaider J, Franklin S. Integer Sparse Distributed Memory. Presented at the 25th Florida Artificial Intelligence Research Society Conference FLAIRS-25. Marco Island, FL; 2012.
8. Manuel A. Your Brain is Cognitive, Not a Database. http://www.saffrontech.com/wp-content/uploads/sites/8/2014/06/Brain-Is-Cognitive.pdf. Accessed August 25, 2015.