Supplemental Material

Table of Contents

I. Image Processing, Quality, and Scanner Effects

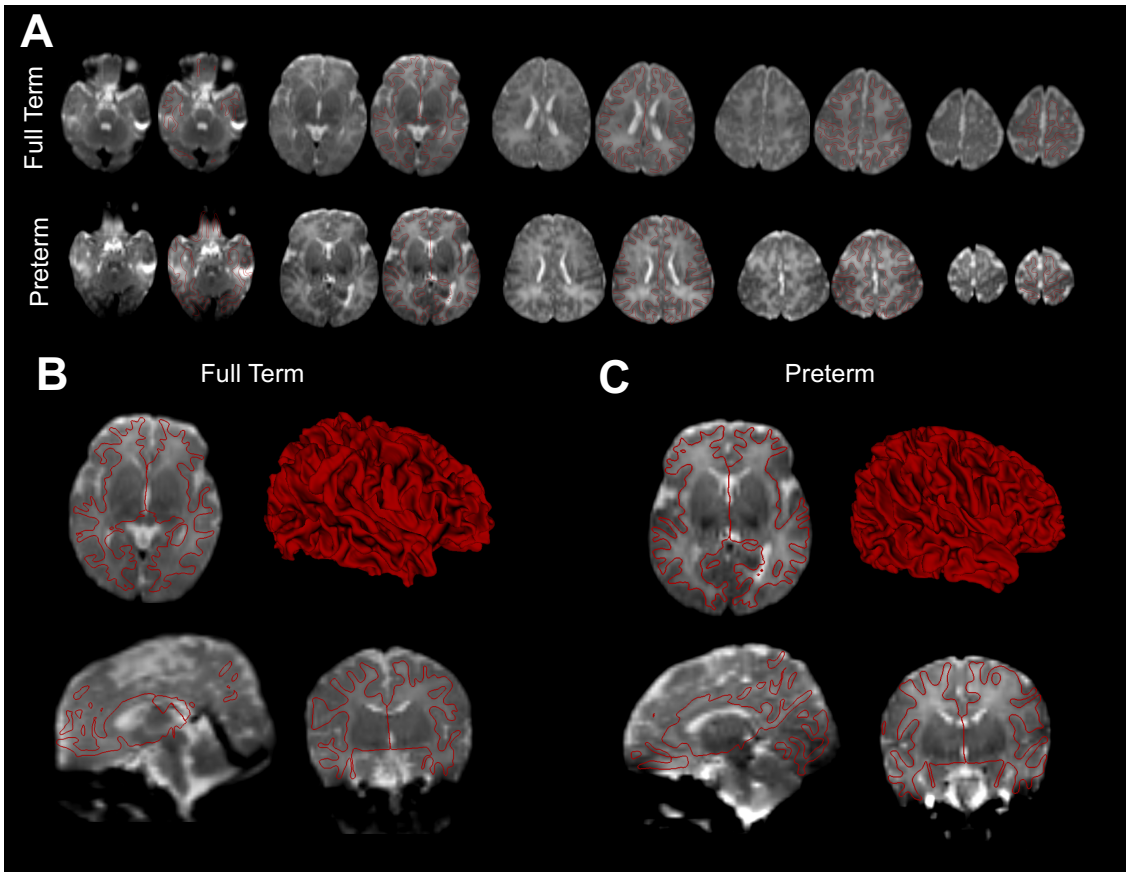
a. Pre-processing and Quality Control

A study-specific, automated quality control (QC) protocol was applied to all raw DWI data. DTIPrep (www.nitrc.org/projects/dtiprep) detected slice-wise and gradient-wise intensity and motion artifacts, removing gradients of poor quality, and corrected for motion and eddy current effects[1]. Next, images were visually inspected in a gradient-wise manner and any additional gradients with artifacts were removed; Supplemental Table 1 presents information regarding the gradients automatically and manually excluded for the full-term and preterm infants. Skull and non-brain tissue were removed using Brain Extraction Tool[2], applied to the average diffusion

baseline image, and tensors were estimated using a weighted least-squares algorithm[3]. All infants with cortical surfaces, T1 images, and DWIs that passed QC (N = 246) were collected. T1 images were registered into DWI space using a rigid followed by deformable registration in ANTS matching the axial diffusivity property maps to the T1 images. The computed transforms were then applied to the cortical surfaces. The surfaces were inspected visually for accurate alignment in 3D Slicer to ensure accuracy in alignment; 219 cases (89%) passed registration QC. Of the 219 which passed QC, 104 infants were lost to follow up and did not have 2-year cognitive data and 3 were excluded due to medical complications, resulting in our finalized dataset of 112 infants (75 full-term, 37 preterm). Example images which passed quality control (both DTIPrep and visual inspection) can be seen in Supplemental Figure 1.

Supplemental Table 1. *Image Quality Summary: Automatic and Manual Gradient Exclusion*

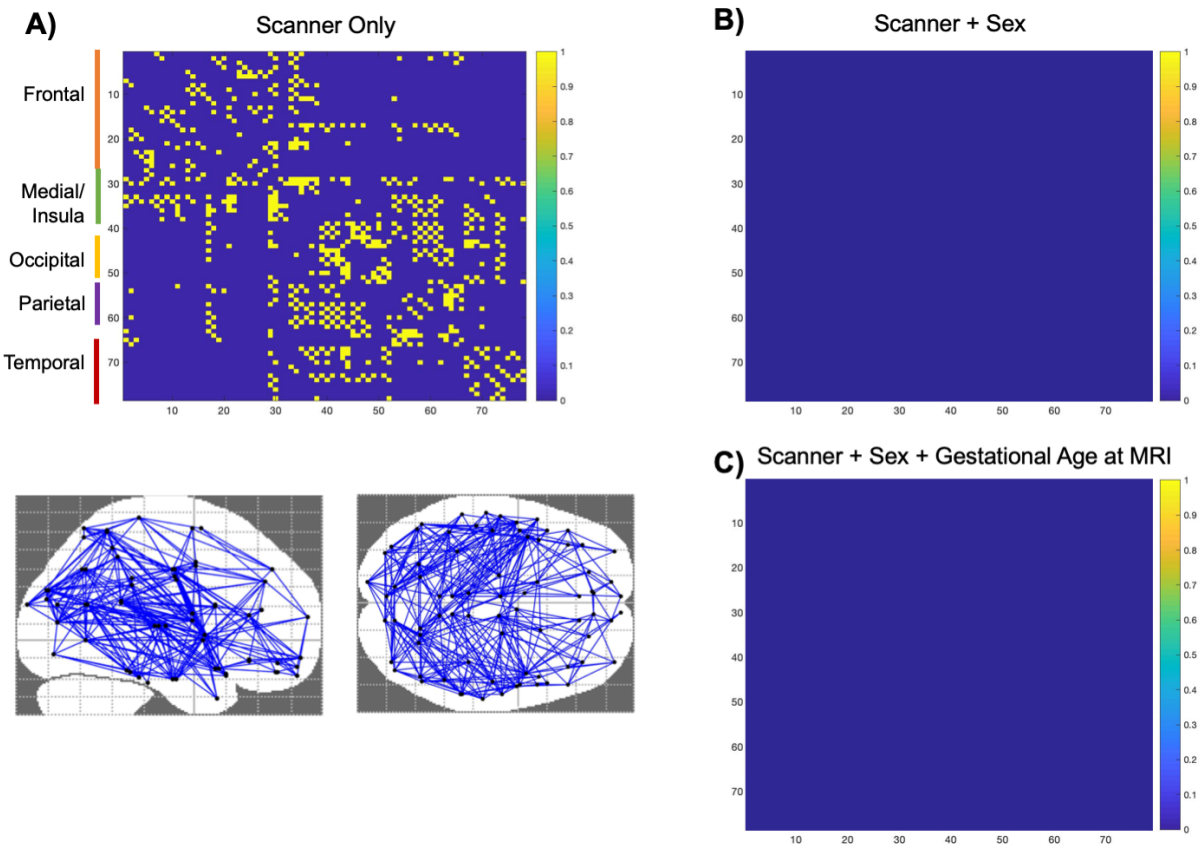
	Full Term		Preterm	
	Mean	*SD*	*Mean*	*SD*
Automatically Excluded	4.03	4.26	4.351	4.856
Visually Excluded	0.15	0.61	0.216	0.886



Supplemental Figure 1. *Representative Neonatal Diffusion Weighted Images Passing QC.* (A) Axial slices for post-processed diffusion weighted images (DWIs) for an infant in the full term (FT; top row) and preterm (PT; bottom row) group are shown, with a left-to-right progression from the ventral to dorsal surface. Each slice is shown alongside the same slice with the co-registered white matter (WM) surface traced by a 2D red line. Axial, sagittal, and coronal slices and the reconstructed WM surface for the same FT (B) and PT participant (C).

b. Testing for scanner effects on the connectome.

DWI images were collected using an identical protocol on either a Siemens Allegra head-only 3T scanner (FT: n = 55, PT n = 27), or a TIM Trio 3T scanner (FT: n = 20, PT n = 10). While we found no evidence to suggest that our results were impacted by group differences in demographic variables between subjects scanned on either scanner (see manuscript for details), we conducted additional sensitivity analyses to determine whether the connectomes generated on either scanner were significantly different. To do this, group differences (Allegra vs. Trio) were tested using the Network Based Statistic (NBS) toolbox (version 1.2, ran using Matlab 2017b). In the NBS toolbox, we tested for significant group differences at the level of individual connections (as these were the features input into the machine learning algorithm) using an FDR correction for multiple comparisons, with 10k permutations used for significance testing and the p-value threshold set to p = 0.001. We tested three models: (1) a two-sample t-test assessing scanner differences, (2) a GLM testing for scanner differences controlling for sex (as there was a significant difference in the distribution of males and females across the scanners), and (3) a GLM testing for scanner differences controlling for sex and gestational age at MRI. The two-sample t-test returned significant results spanning the brain (see Figure 2 below); however, none of these results were significant after adjusting for sex or the combination of sex and gestational age at MRI. We even toggled the p-value threshold to p = 0.01 and p = 0.05 and still found no significant associations after adjusting for sex or age and sex. Based on these findings, we conclude that scanner differences had no major impact on the findings from this study.



Supplemental Figure 2. (A) A two-sample t-test with FDR correction (10k permutations and p-value threshold of p = 0.001) found many connections across the brain differed based on scanner (Allegra vs. Trio). However, after adjusting for sex (B) and sex plus gestational age at MRI (C), no significant findings remained. Adjacency matrices represent connections which were found to significantly relate to scanner (value of 1, yellow) or not (value of 0, blue).

II. ELC 2-year score group classification model training

Both models (classification and prediction) are trained and tested using a 10-fold cross-validation strategy using only FT infant data. The cross-validation strategy first evenly divided (as best as possible) infants into ten folds, where pairs of twins are in the same fold, infants are randomly assigned to each fold, and the ratio of below median (BM) and above median (AM) infants are maintained in each fold. At each iteration, one fold is used to test the model, and the

remaining nine folds are used to train the model. At completion, ten different trained pipelines (classification model and prediction model) have been created.

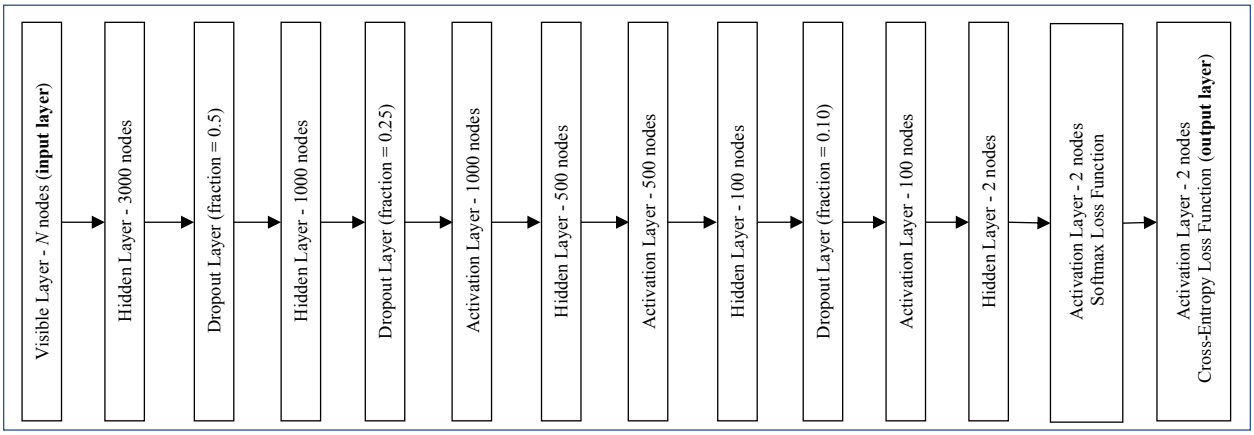
For the ELC group score classification model, the optimal momentum (p_m) and learning rate (p_{lr}) neural network model parameters were found by incorporating a grid search procedure in our cross-validation strategy. Specifically, an independent two-dimension grid-search procedure was performed for each left-out-fold, where the values stored at grid coordinate (p_m, p_{lr}) were the mean and standard deviation classification values. In particular, p_m was adjusted in increments of 0.05 starting at 0.001 and ending at 1.0, while p_{lr} was adjusted in increments of 0.0001 starting at 0.0005 and ending at 0.01. When the grid-search completes, the parameter values that achieved the highest classification accuracy are selected. It should be noted that when the decay value was set to a particularly small value ($\sim 10^{-6}$), it had little to no effect on the classification accuracy, so this model parameter was not included in our grid-search procedure. The momentum and learning rate grid search parameters that yielded the reported accuracy results for each of the 10 classification models were $p_m = 0.015$ and $p_{lr} = 0.001$, respectively.

III. ELC 2-year median score group classification model – first step in pipeline

a. Dense neural network

The classification model is represented by a dense neural network as shown in Supplemental Figure 3, where the hidden-layer architecture was $[3000, 1000, 500, 100, 2]$. One additional supervised learning layer was added when the model was trained that also had two nodes, one for each ELC 2-year median score group (BM and AM). The dense neural network is trained and tested only using connectivity features from FT infants, and based primarily on the small size of the training population, the back-propagation optimization procedure (stochastic gradient-decent algorithm [4]) and the classification loss function (categorical cross-entropy function [5]) were used to compute the optimal edge weight and bias values at each layer. Once the supervised

training step was completed, the supervised training layer was removed, and the nodes in the output layer were used to estimate the ELC 2-year median score classification group and probability values. More specifically, the classification probability value is a real number in [0 1], where a value of one implies the neonate is at top of the BM or AM classification group. For instance, if the model classifies the neonate as AM, and the range of the AM classification group is in [110 150], then a classification probability equal to one would imply the neonate is at or near 150. Similarly, a classification probability of zero would imply the neonate is at or near 110.



Supplemental Figure 3 *The dense neural network design implemented by the 2-year ELC median score group classification model.*

The network architecture included one visible (input) layer that defines N=3003 nodes (the dimension of the connectivity feature vector), five hidden layers, five activation layers, and three dropout layers. The three hidden activation layers used rectified linear unit (ReLU) functions, and the last activation layer (output layer) used the cross-entropy loss function[1] [6,9]. To prevent overfitting [7] the first dropout layer randomly dropped 50% (rate = 0.5) of the nodes in the previous hidden layer (3000 nodes), the second dropout layer randomly dropped 25% (rate = 0.25) of the nodes in the previous hidden layer (1000 nodes), and the third dropout layer randomly

[1] https://www.mathworks.com/help/nnet/ref/classificationlayer.html

dropped 10% (rate = 0.10) of the nodes in the previous hidden layer (100 nodes). Lastly, the MATLAB (see software below) "*classify"* model function (which returns the classification result of the loss function) is applied to the output layer to determine the ELC 2-year *classification group* (BM or AM) and the MATLAB "*predict"* (which returns the value of the loss function) function is applied to the output layer to estimate the *classification probability*.

The software used to develop, train, and test the dense neural network is written in MATLAB using the Deep Learning Toolbox that wrap the C++ NVIDIA CUDA deep neural network libraries (https://developer.nvidia.com/cudnn). All the reported results were executed on a NVIDIA GeForce GTX 970 graphics card that had 4GB of memory.

b. Full-term (FT) classification results

Supplemental Figure 4 summarizes the test-fold results found by the 10-fold cross-validation strategy and our grid search procedure (grid search is incorporated in to strategy to identify the optimal learning rate and momentum deep learning model parameters). In particular, the positive predictive (PPV), negative predictive (NPV), sensitivity, specificity, and accuracy confusion matrix results for each test fold are show in Supplemental Figure 3 (a). Using the results of the ten confusion matrices, the average, standard deviation, and standard error ($n = 75$) values are in Supplemental Figure 3 (b), and the confusion matrix and formula definitions are shown in Supplemental Figure 3 (c).

Fold 1				
	AM	BM		
AM	4	0	100%	4
BM	0	2	100%	2
	100%	100%	6	
	4	2		100%

Fold 2				
	AM	BM		
AM	2	1	67%	3
BM	0	5	100%	5
	100%	83%	8	
	2	6		88%

Fold 3				
	AM	BM		
AM	3	0	100%	3
BM	1	3	75%	4
	75%	100%	7	
	4	3		86%

Fold 4				
	AM	BM		
AM	3	0	100%	3
BM	1	4	80%	5
	75%	100%	8	
	4	4		88%

Fold 5				
	AM	BM		
AM	4	1	80%	5
BM	0	3	100%	3
	100%	75%	8	
	4	4		88%

Fold 6				
	AM	BM		
AM	2	0	100%	2
BM	1	3	75%	4
	67%	100%	6	
	3	3		83%

Fold 7				
	AM	BM		
AM	5	1	83%	6
BM	0	2	100%	2
	100%	67%	8	
	5	3		88%

Fold 8				
	AM	BM		
AM	4	0	100%	4
BM	1	4	80%	5
	80%	100%	9	
	5	4		89%

Fold 9				
	AM	BM		
AM	3	0	100%	3
BM	0	4	100%	4
	100%	100%	7	
	3	4		100%

Fold 10				
	AM	BM		
AM	4	1	80%	5
BM	0	3	100%	3
	100%	75%	8	
	4	4		88%

	AVERAGE	STDEV	STD ERROR
PPV	91.0%	12.4%	0.33%
NPV	91.0%	11.7%	0.31%
Sensitivity	89.7%	13.7%	0.37%
Specificity	90.0%	13.5%	0.36%
Accuracy	89.5%	5.7%	0.15%

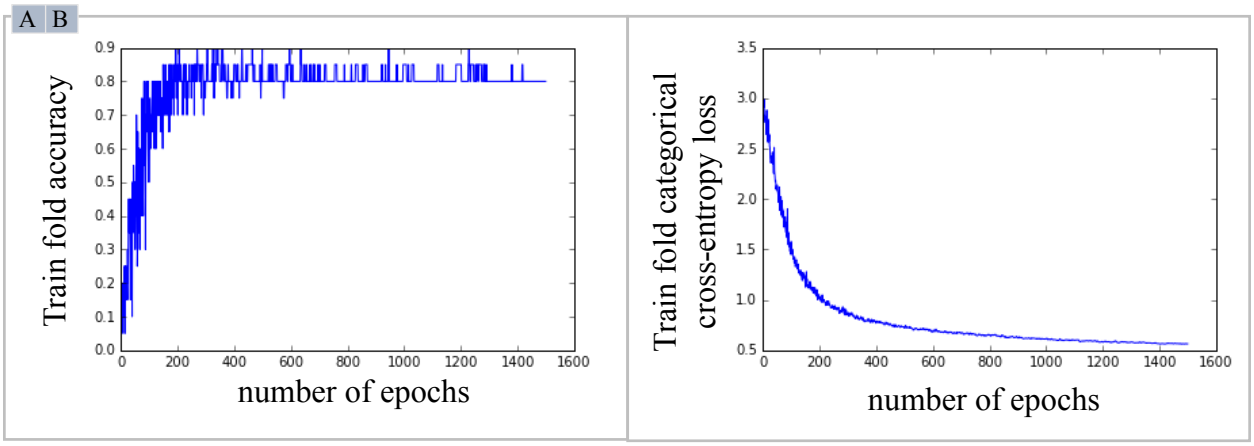
2x2 Confusion Matrix				
	True AM	True BM		
Predicted AM	TP	FP	PPV	TP+FP
Predicted BM	FN	TN	NPV	FN+TN
	Sensitivity	Specificity	TP+FP+FN+TN	
	TP+FN	FP+TN		Accuracy

$$Sensitivity = \frac{TP}{TP + FN} \qquad PPV = \frac{TP}{TP+FP}$$

$$Specificity = \frac{TN}{FP + TN} \qquad NPV = \frac{TN}{FN+TN}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP}$$

Supplemental Figure 4. *Summary of FT ELC 2-year score group classification result via 10-fold cross-validations: Confusion matrices for each test fold (top), average, standard deviation, and standard error values (middle), and the confusion matrix cell and formula definitions (bottom) are shown.*



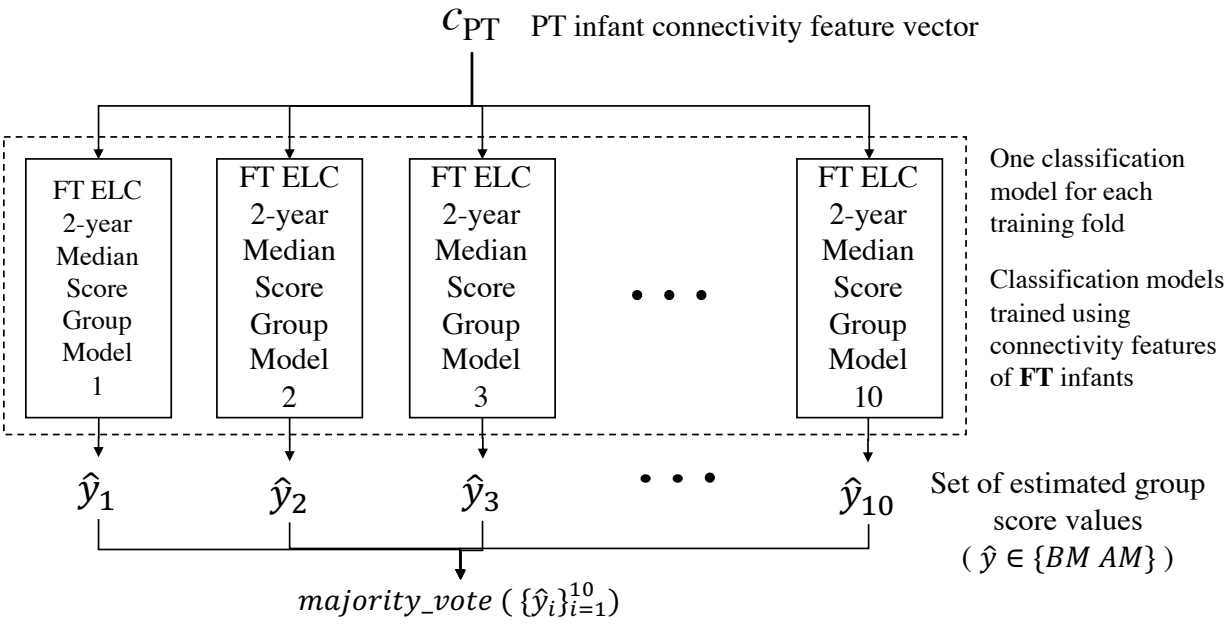
Supplemental Figure 5. *Example training fold classification accuracy and loss plots for one round in the 10-fold cross validation strategy: (A) shows the classification accuracy, and (B) shows the categorical cross-entropy loss.*

Supplemental Figure 5 shows the accuracy and categorical cross-entropy loss vs. number of epoch plots for one typical round in the 10-fold cross-validation strategy (i.e. 1 test fold and 9 training folds). For these plots the number of epochs is extended to 1500, however the stopping criteria used to generate the classification accuracy results reported in Supplemental Figure 3 was determined to computing the average mean square error (MSE), $MSE_e = 1/\delta \sum_{i=1}^{\delta}(l_i^e - l_i^{e-1})^2$ for the last 10 consecutive epochs (i.e., $e = 1,2 \ldots 10$) that were less than 0.01, where l_i^e is the train fold categorical cross-entropy loss epoch e, and l_i^{e-1} is train fold categorical cross-entropy loss at epoch $e - 1$. Based on our experiments, over several different 10-fold simulations the epoch that resulted in a $MSE_e < 0.01$ was consistently reached by epoch 800.

c. Pre-term (PT) Classification results

The ELC 2-year median score group classification approach was also used to identify the score group (BM and AM) of PT infants not used in the training procedure. As shown in Supplemental Figure 5, the 10-fold cross validation strategy creates 10 different classification models, all trained using FT infant connectivity vectors. Given a PT infant connectivity feature vector (c_{PT}), the 10

different classification models are used to estimate a set of score group values, $\{\hat{y}_i\}_{i=1}^{10}$ where $\hat{y}_i$ is the group score of the i^{th} classification model. Next, a majority vote technique is applied to $\{\hat{y}_i\}_{i=1}^{10}$, and the estimated group score with the greatest occurrence is assigned to the PT infant. For example, if the estimated median score group values of a PT infant are $\{AM, AM, BM, AM, AM, AM, BM, BM, AM, BM\}$, then the PT infant is assigned to the above median ELC 2-year median score group. The majority vote classification results for each PT infant in our study is shown in Supplemental Table 2. The score group classification accuracy is 84% (31 of 37 PT infants are assigned to the correct ELC 2-year median score group).



Supplemental Figure 6. *Majority vote PT infant ELC 2-year median score group approach using classification models trained with FT infant connectivity feature vectors. Because the 10-fold cross-validation strategy is used, 10 different classification models were created that are used in the median score group voting process.*

Supplemental Table 2. *PT infant ELC 2-year median score group majority vote results. The score group (BM or AM) of 31 PT infants (out of 37 total PT infants) were assigned to the correct ELC 2-year median score group.*

		Classification Votes			
Preterm Neonate ID	True ELC 2-year median group	BM	AM	Majority	Classification Error
neo-0205-2-1	BM	8	2	BM	
neo-0458-1-1	BM	9	1	BM	
neo-0460-1-1	AM	3	7	AM	
neo-0490-1-1	BM	8	2	BM	
T0162-1-2	BM	9	1	BM	
T0175-1-1	AM	2	8	AM	
T0175-1-2	AM	0	10	AM	
T0178-1-1	AM	2	8	AM	
T0180-1-2	BM	7	3	BM	
T0189-1-2	BM	7	3	BM	
T0190-1-1	AM	3	7	AM	
T0190-1-2	BM	3	7	AM	x
T0191-1-1	AM	2	8	AM	
T0203-1-1	AM	3	7	AM	
T0210-1-1	AM	7	3	BM	x
T0210-1-2	AM	7	3	BM	x
T0223-1-1	AM	3	7	AM	
T0223-1-2	AM	4	6	AM	
T0245-1-1	AM	8	2	BM	x
T0247-1-2	AM	2	8	AM	
T0251-1-1	BM	8	2	BM	
T0252-1-1	AM	2	8	AM	
T0252-1-2	BM	8	2	BM	
T0254-1-1	BM	8	2	BM	
T0254-1-2	BM	9	1	BM	
T0267-1-2	BM	7	3	BM	
T0270-1-1	BM	10	0	BM	
T0270-1-2	AM	4	6	AM	
T0272-1-1	BM	8	2	BM	
T0275-1-1	BM	2	8	AM	x
T0283-1-1	BM	10	0	BM	
T0283-1-2	BM	8	2	BM	
T0286-1-2	BM	7	3	BM	
T0301-1-1	BM	2	8	AM	x
T0301-1-2	BM	8	2	BM	
T0306-1-1	BM	6	4	BM	
T0306-1-2	BM	7	3	BM	

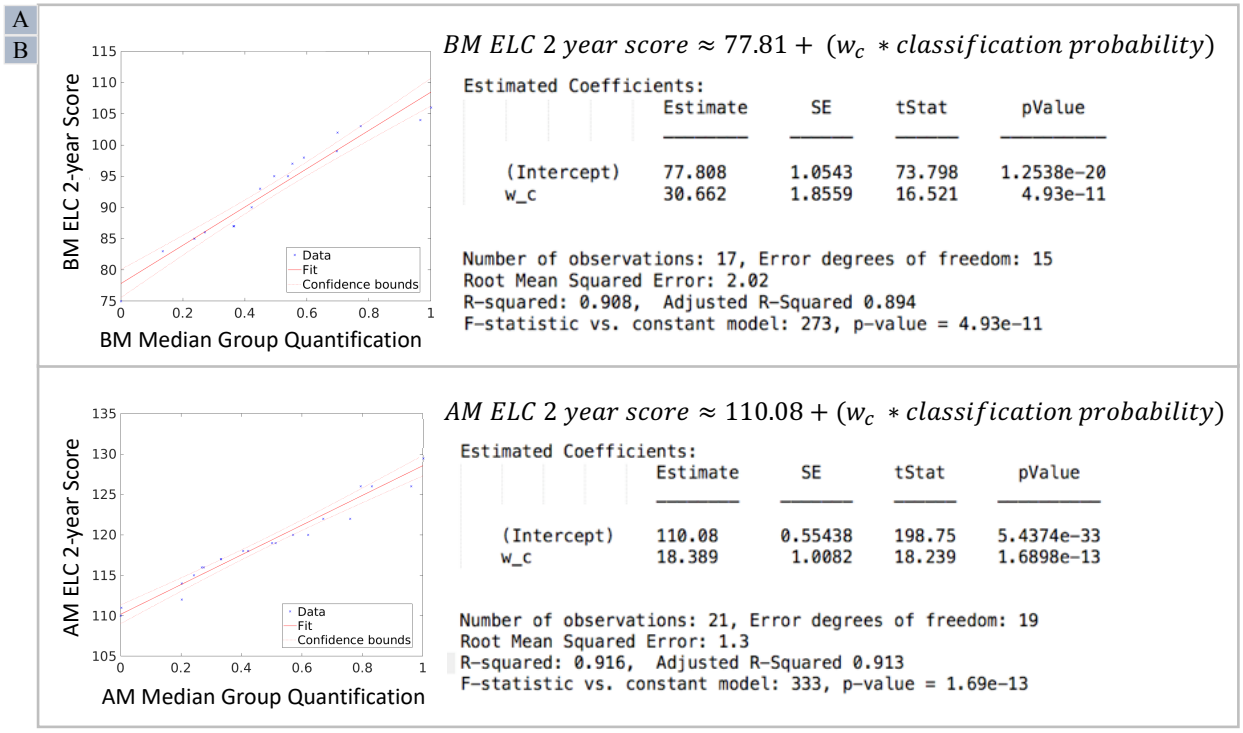
IV. ELC 2-year score prediction model – second step in pipeline

a. Linear regression

For each median score group classification model created by the 10-fold cross-validation strategy shown in Supplemental Figure 7, the classification probability values found by the neural network classifier along with the known ELC 2-year score values are then used to create two fine-tuned linear regression models, one for each median score group (BM and AM), that are trained using an ordinary least squares technique [8]. In general, the form of the linear equation is

$$ELC\ 2\ year\ score = \ intercept + \ w_c * classification\ probability, \qquad \text{Eq. (1)}$$

where classification probability is the predictor (independent) variable found by the output layer in our neural network (see Supplemental Figure 3), and ELC 2-year score is the known response (dependent) variable. Supplemental Figure 7 shows a bank of two ELC 2-year score prediction models, one for each score group, trained using the ELC 2-year classification probability values and the ELC 2-year score values of one training fold. It is important to note, the BM and AM ELC 2-year score predictions models created by each remaining training fold have very similar intercept, weight coefficient (w_c), and R^2 properties.



Supplemental Figure 7. *BM and AM linear regression models found using the ELC 2-year scores and classification probability values for FT infants in a representative training fold: (A) learned BM ELC 2-year score predication model and (B) learned AM ELC 2-year score predication model. The red-line is the ELC 2-year score prediction line fit to the normalized classification probability values estimated by the neural network and known ELC 2-year score values, blue-points represent training data predictor and response values, and the red-dotted line is the 95% confidence interval found by the linear regression model.*

For example, using the two median score group prediction models in Supplemental Figure 7, say the ELC 2-year median classification group and probability value for an infant connectivity dataset was AM and 0.78, respectively. Next to predict the ELC 2-year score, first the AM prediction model is selected, and then the predicted score is calculated $110.08 + (18.39 * 0.78) \approx 124.0$. Since our approach has 10 trained prediction pipelines (classification model followed by prediction model), one for each training fold in the cross-validation strategy, our approach will yield 10 score predictions for infant connectivity datasets not employed in prediction

model generation (such as the PT datasets). Via these 10 predictions we report the ELC 2-year score as mean +/- standard deviation. This provides clinicians with a range of scores instead of just a single ELC 2-year score.

Lastly, it is possible the ELC 2-year median score group assigned to an infant (FT or PT) by the neural network is not correct, e.g. true score group is AM but the infant was classified as BM by the neural network. During prediction model training, only those infants are used that are correctly predicted by the classification network. Obviously, during the test or validation-phase, because the true median score group is not known, the ELC 2-year score is determined by the ELC 2-year score prediction model that is matched to the group predicted by the classification model.

b. FT prediction results

Supplemental Table 3 shows the ELC 2-year score prediction results for each FT infant in our study. It is important to note, since each FT infant can only be in one testing fold, we can only report one score group classification (i.e. no majority vote), and only one prediction score value. In short, the ELC 2-year score is not in mean +/- standard deviation format.

Supplemental Table 3. *FT infant ELC 2-year score prediction results.*

FT Neonate ID	True ELC 2-year median score group	True ELC 2-year score	Classified ELC 2-year median score group	Predicted ELC 2-year score	Absolute Error	Classification Error
neo-0092-3-1	BM	87	BM	84.69	2.31	
neo-0113-2-1	BM	86	BM	87.52	1.52	
neo-0176-2-1	BM	93	BM	89.45	3.55	
neo-0304-2-1	AM	117	AM	114.17	2.83	
neo-0318-2-1	BM	98	BM	93.59	4.41	
neo-0343-2-1	AM	129	AM	131.29	2.29	
neo-0346-2-1	AM	116	AM	118.22	2.22	
neo-0378-1-1	AM	126	AM	123.19	2.81	
neo-0393-1-1	BM	104	BM	101.35	2.65	
neo-0393-2-1	BM	71	BM	67.89	3.11	
neo-0394-1-1	AM	131	AM	132.77	1.77	
neo-0397-1-1	AM	130	AM	135.24	5.24	
neo-0404-1-1	AM	126	AM	131.48	5.48	

neo-0409-1-1	BM	87	BM	88.26	1.26	
neo-0411-1-1	BM	109	AM	113.27	4.27	x
neo-0413-1-1	BM	99	BM	103.47	4.47	
neo-0417-1-1	AM	113	AM	112.71	0.29	
neo-0426-1-1	BM	95	BM	99.8	4.8	
neo-0427-1-1	AM	119	AM	123.41	4.41	
neo-0429-1-1	BM	97	BM	99.83	2.83	
neo-0431-1-1	BM	94	BM	95.91	1.91	
neo-0446-1-1	BM	83	BM	85.2	2.2	
neo-0449-1-1	AM	118	AM	121.67	3.67	
neo-0462-1-1	BM	93	BM	98.53	5.53	
neo-0464-1-1	AM	117	AM	119.8	2.8	
neo-0466-1-1	BM	97	BM	91.23	5.77	
neo-0471-1-1	BM	97	BM	102.43	5.43	
neo-0478-1-1	BM	77	BM	79.48	2.48	
neo-0482-1-1	AM	116	AM	118.5	2.5	
neo-0493-1-1	AM	120	AM	115.49	4.51	
neo-0497-1-1	BM	93	BM	96.73	3.73	
neo-0498-1-1	AM	147	AM	152.5	5.5	
neo-0500-1-1	AM	122	AM	126.59	4.59	
neo-0501-1-1	AM	113	AM	113.48	0.48	
neo-0502-1-1	AM	122	AM	123.08	1.08	
neo-0507-1-1	BM	95	BM	98.92	3.92	
neo-0511-1-1	AM	114	BM	109.47	5.47	x
neo-0519-1-1	AM	115	AM	117.47	2.47	
neo-0522-1-1	BM	103	BM	101.13	1.87	
neo-0523-1-1	BM	87	BM	88.09	1.09	
neo-0524-1-1	BM	95	BM	90.73	4.27	
neo-0528-1-1	AM	126	AM	130.8	4.8	
neo-0530-1-1	AM	114	AM	116.15	2.15	
neo-0534-1-1	BM	99	BM	105.87	6.87	
neo-0537-1-1	AM	119	AM	114.73	4.27	
neo-0546-1-1	AM	123	AM	126.54	3.54	
neo-0562-1-1	AM	129	AM	131.51	2.51	
T0193-1-1	BM	108	AM	113.53	5.53	x
T0193-1-2	AM	110	BM	106.98	3.02	x
T0201-1-1	BM	85	BM	87.29	2.29	
T0204-1-1	BM	97	BM	92.78	4.22	
T0204-1-2	BM	102	BM	105.21	3.21	

T0208-1-2	AM	132	AM	135.36	3.36	
T0214-2-1	BM	81	BM	84.78	3.78	
T0214-2-2	BM	75	BM	79.2	4.2	
T0217-1-1	BM	95	BM	98.34	3.34	
T0222-1-1	AM	116	AM	118.26	2.26	
T0225-1-2	BM	93	BM	88.55	4.45	
T0229-1-2	BM	90	BM	93.55	3.55	
T0233-1-1	AM	113	BM	109.36	4.36	x
T0233-1-2	AM	124	AM	127.28	3.28	
T0237-1-1	AM	120	AM	124.67	4.67	
T0243-1-2	AM	130	AM	134.38	4.38	
T0248-1-1	BM	106	BM	101.36	4.64	
T0249-1-1	BM	107	AM	111.91	4.91	x
T0253-1-1	AM	112	AM	113.51	1.51	
T0253-1-2	AM	114	AM	114.42	0.42	
T0266-1-2	AM	117	AM	119.2	2.2	
T0291-1-2	BM	85	BM	86.33	1.33	
T0293-1-1	BM	106	AM	110.62	4.62	x
T0293-1-2	AM	118	AM	122.65	4.65	
T0294-1-1	AM	121	AM	115.35	5.65	
T0303-1-2	AM	110	BM	106.99	3.01	x
T0310-1-2	BM	94	BM	99.75	5.75	
T0313-1-2	AM	111	AM	112.91	1.91	

c. PT prediction results

Supplemental Table 4 shows the ELC 2-year score prediction results for each PT infant in our study. Since these infants were not included in the 10-fold cross validation strategy, the ELC 2-year scores are in mean +/- standard deviation format. For ELC 2-year score group classification results via majority vote please see Supplemental Table 2. Supplemental Table 5 shows the predicted ELC 2-year score values for each prediction model created by the 10-fold cross-validation strategy. Note: the mean predicted ELC 2-year score and the standard deviations reported in Supplemental Table 4 were computed using the values in Supplemental Table 5.

Supplemental Table 4. *PT infant ELC 2-year score prediction results.*

Preterm Neonate ID	True ELC 2-year median group	True ELC 2-year score	Predicted mean difference ELC 2-year score	Predicted stdev ELC 2-year score
neo-0205-2-1	BM	100	3.23	0.68
neo-0458-1-1	BM	93	5.22	0.79
neo-0460-1-1	AM	116	2.71	1.14
neo-0490-1-1	BM	105	2.79	0.64
T0162-1-2	BM	51	6.21	1.35
T0175-1-1	AM	131	5.54	1.01
T0175-1-2	AM	126	5.57	1.07
T0178-1-1	AM	130	4.80	0.92
T0180-1-2	BM	78	5.54	0.97
T0189-1-2	BM	99	5.27	0.79
T0190-1-1	AM	112	1.20	0.70
T0190-1-2	BM	105	6.51	0.77
T0191-1-1	AM	114	0.44	0.28
T0203-1-1	AM	130	3.97	0.73
T0210-1-1	AM	113	4.92	1.21
T0210-1-2	AM	115	7.30	1.28
T0223-1-1	AM	117	2.38	0.53
T0223-1-2	AM	114	0.95	0.69
T0245-1-1	AM	113	4.38	0.95
T0247-1-2	AM	122	3.95	1.45
T0251-1-1	BM	98	4.03	0.68
T0252-1-1	AM	117	3.40	1.19
T0252-1-2	BM	109	4.39	1.19
T0254-1-1	BM	83	5.81	1.05
T0254-1-2	BM	94	4.55	0.56
T0267-1-2	BM	99	4.84	1.15
T0270-1-1	BM	106	5.29	0.77
T0270-1-2	AM	110	3.07	1.07
T0272-1-1	BM	82	4.91	0.65
T0275-1-1	BM	107	5.21	1.30
T0283-1-1	BM	100	5.10	0.67
T0283-1-2	BM	107	4.92	1.40
T0286-1-2	BM	88	3.90	1.05
T0301-1-1	BM	106	6.66	1.02
T0301-1-2	BM	100	5.43	1.24

T0306-1-1	BM	96	5.75	1.09
T0306-1-2	BM	106	5.15	1.04

Supplemental Table 5. *PT infant ELC 2-year score prediction results for each model created by the 10-fold cross-validation strategy.*

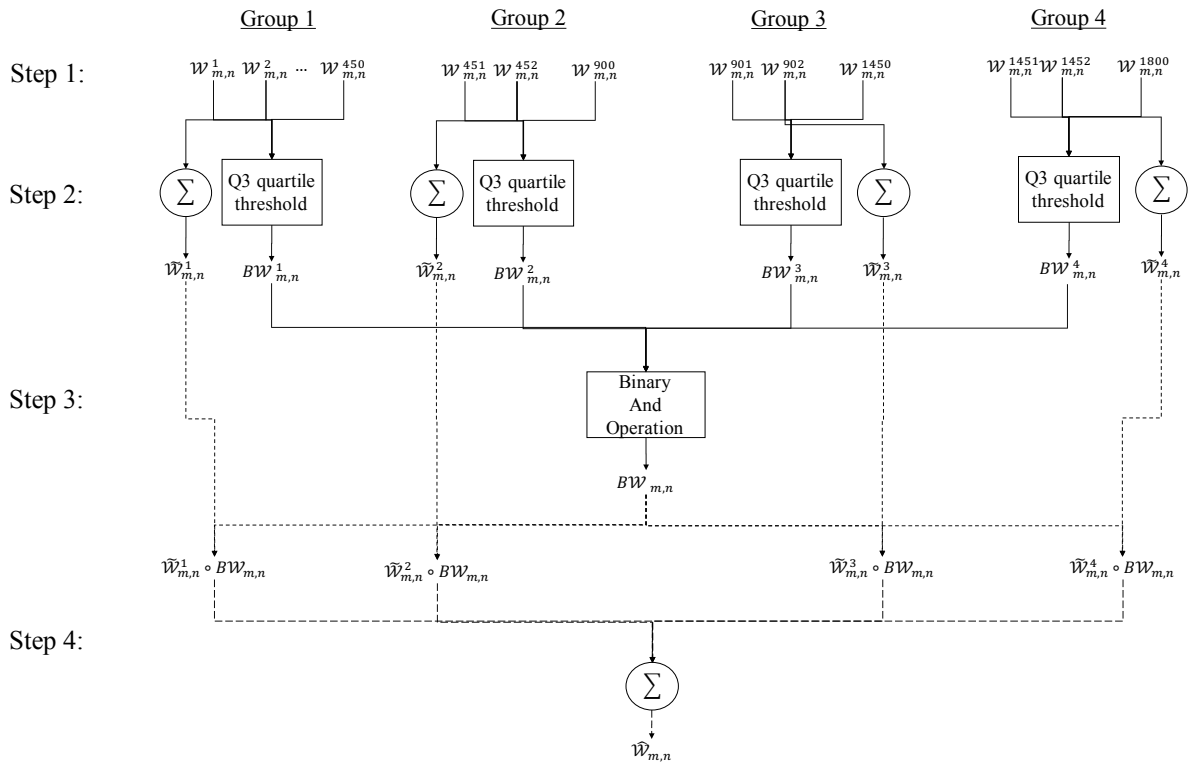
Preterm Neonate ID	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8	Model 9	Model 10
neo-0205-2-1	96.02	97.31	96.25	97.06	96.33	97.38	96.59	95.76	97.91	97.07
neo-0458-1-1	87.08	88.29	87.89	87.53	88.65	88.51	88.69	87.19	86.26	87.71
neo-0460-1-1	113.02	113.89	114.74	113.29	112.53	113.18	111.53	113.67	111.92	115.17
neo-0490-1-1	101.58	103.07	101.67	103.23	101.94	101.81	101.81	102.69	102.66	101.63
T0162-1-2	58.23	56.88	58.32	55.77	58.14	54.66	56.85	57.77	59.07	56.40
T0175-1-1	125.58	127.05	125.21	124.85	125.47	125.94	123.94	125.67	124.11	126.78
T0175-1-2	120.34	119.90	121.66	122.06	118.57	120.20	120.53	119.34	120.22	121.43
T0178-1-1	125.29	125.63	126.27	124.93	123.57	124.62	124.57	126.28	126.32	124.52
T0180-1-2	83.29	83.95	83.32	82.74	82.98	83.59	83.66	85.87	83.82	82.23
T0189-1-2	92.79	92.63	95.07	94.49	93.49	93.79	93.73	94.57	93.72	93.06
T0190-1-1	111.76	114.22	113.63	113.77	111.59	112.55	113.30	112.72	113.18	113.98
T0190-1-2	110.89	111.94	111.87	112.54	110.03	112.19	111.50	112.07	111.35	110.75
T0191-1-1	113.49	113.40	113.33	113.75	114.09	114.37	115.08	113.74	114.31	114.29
T0203-1-1	125.89	126.29	125.97	125.69	127.00	125.38	125.44	124.89	126.66	127.11
T0210-1-1	107.86	108.44	107.86	109.91	109.62	108.72	107.04	105.99	106.93	108.40
T0210-1-2	107.23	107.50	107.63	110.67	107.74	106.50	105.72	108.06	107.77	108.14
T0223-1-1	115.32	114.45	114.37	115.51	113.72	114.96	114.71	114.29	114.23	114.62
T0223-1-2	111.90	113.18	113.29	113.95	112.84	115.02	114.98	113.84	111.97	113.57
T0245-1-1	107.98	108.16	109.25	106.86	109.89	109.07	109.20	107.50	108.93	109.36
T0247-1-2	119.14	121.44	116.87	117.49	116.58	117.63	117.37	117.00	118.22	118.81
T0251-1-1	94.32	93.29	93.82	94.37	94.01	94.78	92.71	93.59	94.96	93.88
T0252-1-1	111.62	114.22	114.92	113.36	114.26	113.71	113.37	111.65	115.11	113.76
T0252-1-2	110.89	113.29	115.45	113.87	114.34	113.20	112.56	112.91	113.63	113.77
T0254-1-1	87.96	89.53	86.90	89.30	89.69	89.45	88.18	87.94	90.37	88.81
T0254-1-2	90.26	89.29	88.65	89.04	89.39	90.27	89.56	88.87	89.93	89.19
T0267-1-2	94.94	92.39	94.07	95.10	95.83	94.65	93.83	92.80	92.93	95.07
T0270-1-1	110.56	111.70	111.03	112.52	111.04	112.28	110.46	111.73	111.38	110.24
T0270-1-2	112.09	113.56	114.57	113.93	113.89	111.55	111.58	112.45	113.56	113.53
T0272-1-1	87.78	86.49	86.64	87.89	85.71	86.38	87.00	87.18	87.06	86.94
T0275-1-1	112.29	114.36	112.24	111.91	114.37	111.53	112.20	111.82	111.20	110.18
T0283-1-1	94.18	96.25	95.43	94.91	94.71	94.12	95.29	94.94	94.98	94.14
T0283-1-2	112.14	112.41	110.14	113.12	113.54	110.32	110.65	113.51	110.39	112.98

T0286-1-2	83.92	85.16	82.19	84.78	84.65	83.81	85.15	83.04	85.19	83.15
T0301-1-1	111.58	112.53	112.84	111.70	112.49	114.18	113.28	114.26	111.25	112.47
T0301-1-2	92.69	94.53	94.15	95.44	93.91	93.07	94.97	96.98	95.44	94.51
T0306-1-1	90.67	91.31	89.36	90.14	91.92	88.39	90.47	90.93	88.93	90.36
T0306-1-2	110.86	109.52	110.38	110.77	111.03	112.05	111.43	111.55	110.57	113.37

V. Connectivity feature selection

a. Sparse hidden layer edge weight matrix

Using the proposed hidden layer architecture design, and the four step process in shown in Supplemental Figure 7, a set of sparse edge weight matrices $\widehat{\mathcal{W}} = \{\widehat{\mathcal{W}}_{3,100}, \widehat{\mathcal{W}}_{100,500}, \widehat{\mathcal{W}}_{50,1000}, \widehat{\mathcal{W}}_{1000,3000}, \widehat{\mathcal{W}}_{3000,N}\}$ are computed based on four independent training groups. Furthermore, each edge weight matrix in $\widehat{\mathcal{W}}$ only includes edge weight values that are in the top 75th percentile. As illustrated in Supplemental Figure 7 a set of 1800 neural networks $\{\mathcal{N}_i\}_{i=1}^{2000}$ are first created by repeating the 10-fold cross-validation training process 200 times, where $\mathcal{N}_i = \{\mathcal{W}_{3,100}^i, \mathcal{W}_{100,500}^i, \mathcal{W}_{500,1000}^i, \mathcal{W}_{1000,3000}^i, \mathcal{W}_{3000,N}^i\}$ is the set of trained dense edge weight matrices. In general, $\mathcal{W}_{m,n}^i$ is a dense edge weight matrix that connects the m-node upper layer to the n-node lower layer.



Supplemental Figure 8. *Four step procedure to compute the sparse edge weight matrix that connect two dense layers in the proposed network architecture design.*

- Step-1: The edge weight matrices $\{\mathcal{W}_{m,n}^{i}\}_{i=1}^{2000}$ that connect the m-node layer to the n-node layer are separated into four different groups that each have 500 edge weight matrices.
- Step 2: A group edge weight matrix $\widehat{\mathcal{W}}_{m,n}^{g} = \sum_{i=1}^{500} \mathcal{W}_{m,n}^{(g-1)*500+i}$ is created by adding all the edge weight matrices in group g. Next, a binary group mask $B\mathcal{W}_{m,n}^{g}$ is created,

$$B\mathcal{W}_{m,n}^{g}(i,j) = \begin{cases} 1 & \widehat{\mathcal{W}}_{m,n}^{g}(i,j) > Q3(\widehat{\mathcal{W}}_{m,n}^{g}) \\ 0 & otherwise \end{cases}, \qquad \text{Eq. (2)}$$

  where $Q3(\widehat{\mathcal{W}}_{m,n}^{g})$ is the 75th percentile based on each value in $\widehat{\mathcal{W}}_{m,n}^{g}$.
- Step 3: A combined binary mask is computed $B\mathcal{W}_{m,n}$ by sequentially performing a binary "and" operation using the binary mask of each group.
- Step 4: A sparse edge weight matrix

$$\widehat{\mathcal{W}}_{m,n} = \sum_{g=1}^{4} \widehat{\mathcal{W}}_{m,n}^{g} \circ B\mathcal{W}_{m,n} \,, \qquad \text{Eq. (3)}$$

is created by performing an element-wise matrix multiplication between the group edge weight matrix and the sparse combined binary mask. This is repeated for each group and added together.

b. Backtrack algorithm

Given the set of sparse edge weight matrices $\widehat{\mathcal{W}}$, the backtrack technique outlined in Supplemental Algorithm ***1*** is performed. When the algorithm completes an N-dimension input feature weight vector $\boldsymbol{w} = (w_1, w_2, \ldots, w_i, \ldots, w_N)$ is returned, where the value of w_i is in [0 1], and a value of one implies connectivity feature f_i has the greatest contribution to score group classification accuracy across a set of 1800 trained neural networks, whereas a value of zero implies connectivity feature f_i has the least (or no) contribution to score group classification accuracy across a set of 2000 trained neural networks**.** In general, Supplemental Algorithm ***1*** works backwards through the sparse edge weight matrices starting at the output layer that has 3 nodes (one for each ELC score group) and completing at the input layer that has N nodes, i.e. one for each connectivity feature.

In particular, on *line-1* $\boldsymbol{w}$ is a 100-dimension row vector that contains the sum of each row in $\widehat{\mathcal{W}}_{20,100}$ which is then normalized by the maximum value in $\boldsymbol{w}$ on *line-2*. A weight value of zero means edge weight values that connect the lower layer nodes to each of the 3 nodes in the upper layer *was never* in the top 75th percentile, which implies it had no contribution to score group classification accuracy. Conversely, a weight value of one means the edge weight values that connect the lower layer nodes to each of the 3 upper layer nodes *were typically (if not always)* in the top 75th percentile, which implies it had the greatest contribution to score group classification accuracy. The loop on *lines-3 to 6*, then propagates the normalized weight values in the 100-dimension row vector to the next lower layer (in this case, the layer that has 500 hidden nodes) by performing an element-wise matrix multiplication, and then essentially repeats *lines-2* and *3* in *lines-5* and *6*. The iterative approach completes when the normalized weight values in the 3000-dimension row vector is propagated to the input layer that has N nodes. Now the values in the N-dimension weight vector $\boldsymbol{w}$ represent a hierarchical, linear combination, of lower hidden layer

nodes that are strongly connected, i.e. edge weight values are in the top 75th percentile, to the last layer that has 3 nodes.

```
Algorithm: Backtrack( 𝒲̂ )

//
// Input:
// 𝒲̂ = ordered set of 75th percent sparse edge weight matrices
// Output:
//   w = N dimension vector of normalized input layer node weights

1. w = sum(𝒲̂{1}, 1 )      // perform row-wise summation, w is a n-dimension vector

2. w = w ./ max( w )       // normalize by the maximum weight value (new values
                           // in [0 1])

3. for i=2:length(𝒲̂)

4.     M = repmat( w', size(𝒲̂{i}, 1 ), 1 )     // take transpose of weight vector and
                                               // use replicated copies to create a
                                               // matrix that has the same dimension as
                                               // the next lower layer sparse edge weight
                                               // matrix at position i

5.     w = sum(𝒲̂{i}.*M, 2 )                    // perform element-wise matrix multiplication
                                               // and then a row-wise summation

6.     w = w ./ max( w )                       // normalize by maximum weight value

   end
```

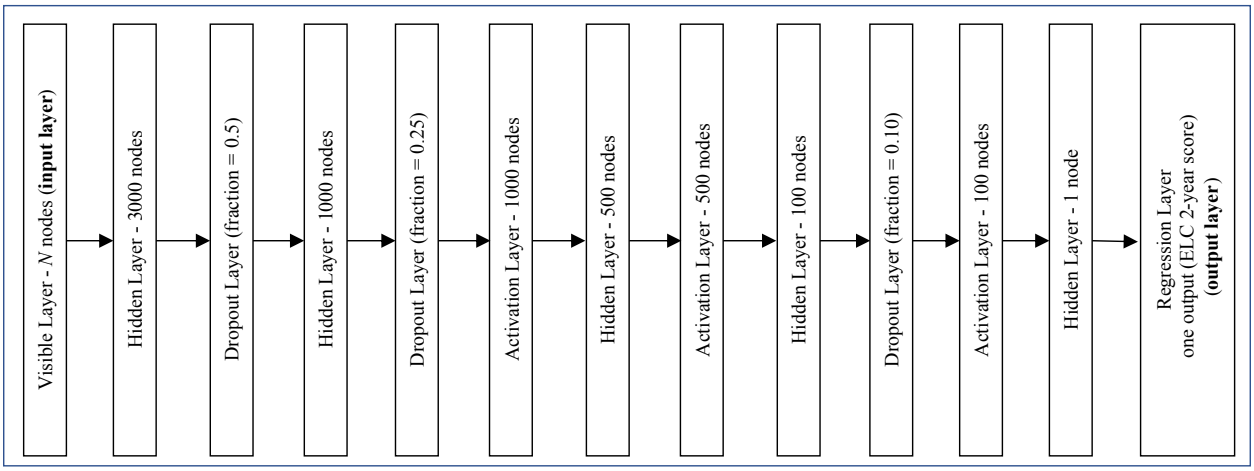
Supplemental Algorithm 1. *Pseudocode (in Matlab like notation) that outlines backtrack approach used to identify those connectivity features at the input layer that are likely to have the greatest contribution to score group classification accuracy at the output layer in the trained neural network.*

c. Weight vector dimension reduction

When the backtrack algorithm completes the values in the N-dimension weight vector $\boldsymbol{w}$ identifies connectivity features that are likely to have the greatest influence on ELC 2-year score group classification accuracy in the trained neural network. For example, if weight values for $w_i, w_j,$ and w_k are all equal to 1, then connectivity features f_i , f_j , and f_k are likely to have the greatest influence. To further reduce the number of non-zero weight values that account for the top 20% of the total weight are selected. The connectivity features that are associated with these weights are then used to construct the connectivity fingerprint.

VI. ELC 2-year single neural network prediction results

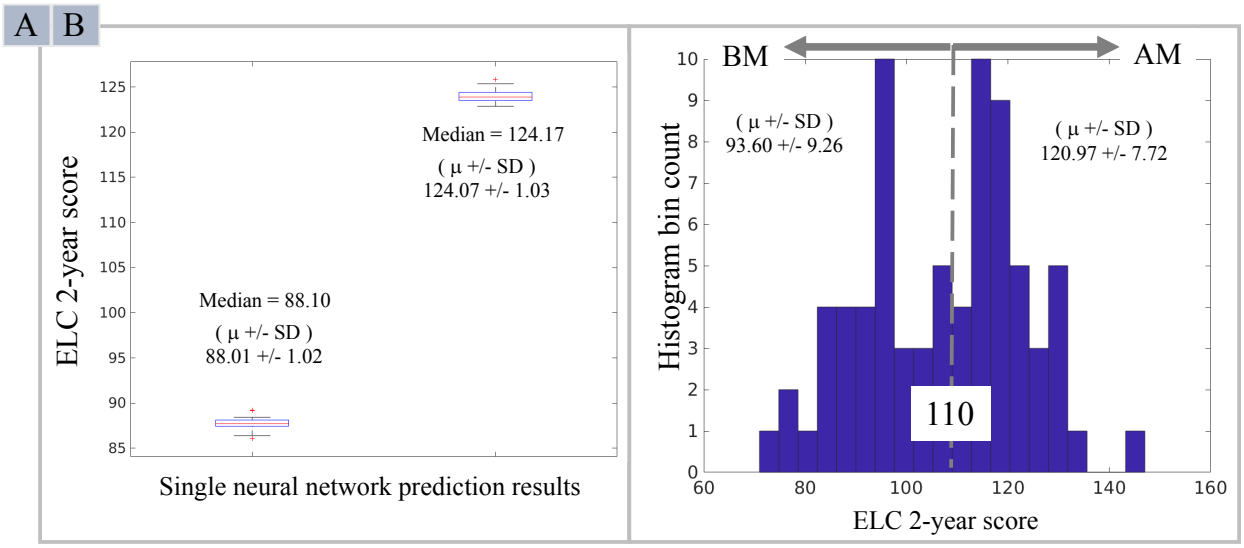
To better understand the rational for the proposed two-step design, the classification neural network (Supplemental Figure 2) was modified as shown in Supplemental Figure 9 to predict a continuous ELC 2-year score. Notably, the last hidden layer was modified to have only one node (that represents a predictor variable) and linear regression prediction layer (i.e. output layer) replaced the softmax and cross-entropy classification layers.



Supplemental Figure 9. *The implementation of the dense single neural network ELC 2-year score prediction model.*

This single neural network prediction model was trained using the 10-fold cross-validation approach (with integrated grid search) described in Supplemental Section II, however the training labels were the actual ELC 2-year scores. The ELC 2-year cross-validation test prediction results are shown Supplemental Figure 10A. In particular, the single neural network design consistently predicts the ELC 2-year score as either 88.01 (+/- 1.02) or 124.07 (+/- 1.03). This result strongly suggests that predicting one continuous measure using high dimension connectivity feature vectors is too complex, and as a result, the single neural network design is overfit to the full-term neonate ELC 2-year score bimodal distribution shown in Figure 10B (Figure 2 in manuscript). In general, this single model result indicates the neural network is better suited to solve a simpler machine learning problem, such as above or below median ELC 2-year median score classification. Moreover, the single model prediction results also suggest these two machine learning problems (ELC 2-year median score classification and ELC 2-year score prediction) are likely not independent (Figure 3 in manuscript), and the models could be sequentially arranged (i.e. output of one model is input to the next model) to form a two-step pipeline design that is capable of

solving two separate, but related, problems with greater accuracy. This important observation was the primary motivation for our two-step design.



Supplemental Figure 10. *Rational for two-step approach: (A) Single neural network model that consistently predicts the ELC 2-year score centered at 88.01 (+/- 1.02) or 124.07 (+/- 1.03), and (B) full-term neonate ELC 2-year distribution (actual scores), where the first mode is centered at 93.60 (+/- 9.26) and the second mode is centered at 120.97 (+/- 7.72). This suggests the single neural network model is overfit to the score distribution. Importantly, this observation suggests these two machine learning problems (classification and prediction) are likely not independent and the models could be sequentially arranged to increase prediction accuracy.*

References

1. Oguz, I., Farzinfar, M., Matsui, J., Budin, F., Liu, Z., Gerig, G., et al. (2014). DTIPrep: quality control of diffusion-weighted images. *Frontiers in Neuroinformatics*, *8*, 4. http://doi.org/10.3389/fninf.2014.00004
2. Smith, S. M. (2002). Fast robust automated brain extraction. *Human Brain Mapping*, *17*(3), 143–155. http://doi.org/10.1002/hbm.10062
3. Goodlett, C. B., Fletcher, P. T., Gilmore, J. H., & Gerig, G. (2009). Group analysis of DTI fiber tract statistics with application to neurodevelopment. *NeuroImage*, *45*(1 Suppl), S133–42. http://doi.org/10.1016/j.neuroimage.2008.10.060
4. Bottou, L. (1991). Stochastic gradient learning in neural networks. Proceedings of Neuro-Nımes, 91(8), 0.

5. Joe, H. (1989). Relative entropy measures of multivariate dependence. Journal of the American Statistical Association, 84(405), 157-164.
6. Dunne, R. A., & Campbell, N. A. (1997). On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In Proc. 8th Aust. Conf. on the Neural Networks, Melbourne (Vol. 181, p. 185).
7. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.
8. Lapin, L. L. (1989). Probability and statistics for modern engineering, Waveland Press, 2nd Edition, ISBN: 0881339962
9. Bishop, C.B. (2006). Pattern Recognition and Machine Learning, Springer-Verlag Berlin, Heidelberg, ISBN: 0387310738