**Laboratory Methods**

**DNA extraction and genome-wide methylation assay:** The DNA samples were extracted from dried neonatal blood spots using the Gentra Puregene® Blood Kit (Qiagen, Hilden, Germany) based on the manufacturer’s protocol. Bisulfite conversion was performed using 500 ng of DNA using EZ DNA methylation kit from Zymo Research Company (Orange CA) according to the company’s instructions. Infinium methylation beadChip arrays were used to perform genome-wide methylation analysis. The assay evaluates >450,000 cytosine (‘CpG’) loci throughout the genome. The CpG methylation sites are distributed throughout the genomic regions including, 5’ and 3’ untranslated regions, translation start sites at 200 and 1500, promoter regions, exon and intronic regions of the genes. Fluorescently labeled beadChips were scanned using Illumina iScanner (Illumina, Inc.). Beta (β) values, an estimation of the methylation level based on the ratio of signal intensities between methylated and unmethylated cytosine at the same locus (alleles) were calculated. Data preprocessing, quality control and background intensity correction were performed to obtain methylated and unmethylated ratio signal intensities. Statistical and bioinformatic analyses were then performed as outlined below.

**Validation of CpG sites using bisulfite pyrosequencing:** To verify the differential methylation status of CpG sites and to determine whether the beadChip hybridizations represented true methylation differences and not artifacts, we performed pyrosequencing. The EZ DNA methylation kit from Zymo Research Company, was used. The 10 CpG markers meeting a stringent threshold to define significant methylation difference in cases versus controls (i.e. p-value <5X10-8) (1) were chosen for validation using appropriate oligos. PyroMark Q24 System was used to screen the methylation sequence.

**Artificial Intelligence Analysis:**

**Deep Learning (DL) analysis.** The β-value of each CpG locus was logged and scaled by auto scaling using its standard deviation. To reduce sample to sample variation, quantile normalization was performed. The hidden first layer was activated by providing sample input to the first layer based on the best parameters. By updating the weights and biases for each layer, all remaining layers were processed. Back propagation was used to regulate the parameters for all hidden layers. To assign new labels to the samples, softmax classifier was used and to tune the parameters of DL model, h2o package of R module was used (2, 3).

**Other machine learning algorithms.** Five other machine learning algorithms : Support vector machine (SVM), Generalized linear Model (GLM), Prediction Analysis for Microarrays (PAM), Random Forest (RF) and Linear Discriminant Analysis (LDA) (2) were used for CoA prediction. To obtain optimal predictive performance and to tune the parameters in the models, caret package of R module was used (4). The pROC package of R was used to compute AUC.

* 1. **Data Set**

There are two groups of analyses performed in this study. First analysis consists of 68 epigenomic markers (individual marker significance defined as FDR p-value < 0.05), and the second uses a group of 10 markers (individual significance defined as p< 5 x10 -8).

* 1. **Data Preparation**

Missing patient values were identified and replaced with a value (the half of the minimum positive values in the original data) used to represent the detection limit. The basis of this assumption is that most missing values are caused by low level of methylation (i.e. below the detection limit). The sample normalization allows general-purpose adjustment for differences among the specimens; data transformation and scaling are two different approaches for making individual features (markers) more comparable. In this data set, the log value of each β-value was centered by its mean (x̅) and auto scaled by its standard deviation(s). Quantile normalization method was used to reduce sample-to-sample variation.

* 1. **Artificial Intelligence Algorithms**

**Deep Learning (DL):** Classical machine learning techniques make predictions directly from a set of features that have been prespecified by the user, however, there is another type of machine learning technique called feature learning. Feature learning (representation learning) refers to techniques that enables the system to automatically determine representations required for feature detection or feature classification from large amounts of raw data.Feature learning algorithms, representation learning algorithms, attempt to preserve input information but transform it in such a way that it can be used as a pre-processing step for prediction of outcomes of interest.Deep-learning methods are representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher level. With the compilation of enough such transformations, very complex functions can be learned. For classification tasks, higher layers of representation further amplifies aspects of the input that are important for discrimination while suppressing irrelevant variations. This type of hierarchical learning process is very powerful as it allows a system to comprehend and learn complex representations directly from the raw data (5), making it useful in many disciplines (6).

**Random Forest (RF):** Recently, ensemble learning methods which generate many classifiers and aggregate their results are very popular. One of the well-known ensemble learning methods is bagging (7) of classification trees. In bagging, successive trees do not depend on earlier trees - each is independently constructed using a different sample of the data set. Random forests add an additional layer of randomness to bagging (8). In addition to constructing each tree using a different sample of the data, random forests change how the classification or regression trees are constructed. In standard trees, each node is split using the best split among all variables. In a random forest, each node is split using the best among a subset of predictors randomly chosen at that node. This strategy performs very well compared to many other classifiers and is robust against overfitting (8). In addition, it has only two parameters (the number of variables in the random subset at each node and the number of trees in the forest) and is usually not very sensitive to or influenced by their values.

**Support vector machine (SVM):** SVMs (9) are a relatively new type of learning algorithm. Their very robust performance with respect to sparse and noisy data makes them the system of choice in a number of applications from text categorization to the field of bioinformatics. When used for classification, they separate a given set of binary labeled training data with a hyper-plane that creates maximal distance between groups (known as ‘the maximal margin hyper-plane’) (10). For cases in which no linear separation is possible, combination with the technique of ‘kernels’, that automatically realizes a non-linear mapping to a feature space can be employed.

**Linear Discriminant Analysis (LDA):** Linear Discriminant Analysis (LDA) is one of the techniques used for data classification and dimensionality reduction. LDA easily handles circumstances where the within-class frequencies are unequal. The method maximizes the ratio of between-class variance to the within-class variance in any particular data set thereby guaranteeing maximal separability (11).

**Prediction Analysis for Microarrays (PAM):** is a statistical technique for class prediction from gene expression data using nearest shrunken centroids. This method identifies the subsets of genes that best characterize each disease.

**Generalized Linear Model (GLM):** The generalized linear models (GLMs) are an extension or generalization of ‘simple’ linear regression. They are a broad class of models that include linear regression, ANOVA, Poisson regression, log-linear models etc. In contrast to multiple regression the distribution of the response/dependent variable can be non-normal nor does it have to be continuous. The response variable value can be predicted from a linear combination of predictor values that are connected to the response variable by a link function. However, there are some limitations to GLM, such as it can have only a linear predictor, and all the predictors must be independent.

* 1. **Software Tools**

We used h2o R package (3) (<https://cran.r-project.org/web/packages/h2o/h2o.pdf>, Author The H2O.ai team Maintainer Tom Kraljevic <tomk@0xdata.com>) to tune the parameters of the DL model and the caret R package (<https://cran.r-project.org/web/packages/caret/caret.pdf>, Maintainer Max Kuhn <mxkuhn@gmail.com> , December 10, 2017 ) to tune the parameters in the other artificial intelligence models (12).

The variable importance functions *varimp* in h2o and *varImp* in caret R packages were utilized to rank the models features in each of the predictive algorithms.

We used pROC R package to compute area under the curve (AUC) of a receiver-operating characteristic (ROC) curve, specificity and sensitivity to assess the overall performance of the models (13).

* 1. **Modeling & Evaluation**

The data were split into training and testing sets to train the model first with a portion of the data and then test the model on the remaining portion. We split the dataset as follows: 80% training and 20% testing since this ratio is generally used in medium size data sets. 10-fold cross validation on the 80% training data was performed during the model construction process. In addition, the process was repeated ten times and the average AUC, sensitivity, specificity and 95% confidence intervals for the test set were calculated.

Several parameters were used to tune the models while implementing them: Number of trees for RF, classification cost for SVM, threshold amount for shrinking toward the centroid for PAM, and for DL model: a) Epochs (number of passes of the full training set), b) l1 (penalty to converge the weights of the model to 0), c) l2 (penalty to prevent the enlargement of the weights), d) input dropout ratio (ratio of ignored neurons in the input layer during training), e) number of hidden layers. In addition to, l1 and l2 parameters, *input\_dropout\_ratio* was used as the third parameter to avoid overfitting in DL model which controls the amount of input layer neurons that are randomly dropped (set to zero), controls overfitting with respect to the input data (useful for high-dimensional noisy data). The key objective was to randomly drop units (along with their connections) from the neural network during training (14). This prevents units from co-adapting too much. Using these approaches, we avoided the biggest risk involved in DL analysis i.e. overfitting (14).

* 1. **Ranking Important Features**

The contribution of a feature (predictor) to the model performance is determined using a model-based approach. We ranked the importance of the features for each of the predictive AI algorithms by using the variable importance functions *varimp* in h2o and *varImp* in caret R packages.

**CoA prediction based on Artificial Intelligence analyses.** CpG marker methylation data were entered in the AI programs and the AUC (95% CI), sensitivity and specificity of marker combinations for CoA detection determined. Each of these CpG loci that were entered had FDR p-value < 0.05. CoA prediction was repeated but this time the CpG loci entered were limited to those that met a stringent threshold for defining significant individual methylation difference – CoA versus controls i.e. p-values <5x10-8. This threshold is recommended to ensure repeatability and generalizability of results in genome-wide association studies (GWAS) studies (1).

**Further Statistical Analysis.** Following acquisition, data were exported to MetaboAnalyst (v4.0), subsequently, quantile normalized, and auto-scaled data were analyzed using principal component analysis (PCA) and partial least-squares discriminant analysis (PLS-DA). Variable importance in the projection (VIP) plots were produced for each pair-wise analysis to determine which genes (CpG loci methylation levels) were most responsible for the separation between case and control groups. Permutation testing was performed (2,000 iterations) to determine if the observed separation in the PLS-DA scores plots was due to chance (p < 0.05). Logistic regression analysis was performed using the Biomarker function in MetaboAnalyst (v4.0). Least Absolute Shrinkage and Selection Operator (LASSO) and stepwise variable selection were utilized for optimizing all predictive model components. The logistic regression model based on LASSO selected genes developed with a 10-fold cross-validation. The area under the receiver operating characteristics curve (AUROC or AUC) was calculated using previously described techniques (15). Sensitivity and specificity values were also calculated.

**Gene ontology and Pathway enrichment analysis.** CpG loci found to be differentially methylated (defined as individual FDR p-value ≤0.05) were analyzed using QIAGEN’S Ingenuity Pathway Analysis (IPA) software. Over-represented molecular and disease pathways and thus altered biological functions or regulatory networks in CoA were identified. All CpG loci without mapping IDs in IPA (HG19) were excluded from analysis. Only genes for which Entrez identifiers were available, were analyzed.

**References**

1. Jannot AS, Ehret G, Perneger T. P < 5 x 10(-8) has emerged as a standard of statistical significance for genome-wide association studies. J Clin Epidemiol. 2015;68(4):460-5.

2. Alakwaa FM, Chaudhary K, Garmire LX. Deep Learning Accurately Predicts Estrogen Receptor Status in Breast Cancer Metabolomics Data. J Proteome Res. 2018;17(1):337-47.

3. Candel A, Parmar V, LeDell E, Arora A. Deep Learning with H2O2018.

4. Kuhn M. Building Predictive Models in R Using the caret Package. . Journal of Statistical Software. 2008;28(5):1-26.

5. Bengio Y. Learning Deep Architectures for AI. Found Trends Mach Learn. 2009;2(1):1-127.

6. Goodfellow I, Bengio Y, Courville A. Deep Learning. MIT Press. 2016.

7. Breiman L. Bagging predictors. Machine Learning. 1996;24(2):123-40.

8. Breiman L. Random Forests. Machine Learning. 2001;45(1):5-32.

9. Cristianini N, Shawe-Taylor J. An introduction to support Vector Machines: and other kernel-based learning methods: Cambridge University Press; 2000. 189 p.

10. Boser BE, Guyon IM, Vapnik VN. A Training Algorithm for Optimal Margin Classifiers: ACM Press; 1992.

11. Balakrishnama S, Ganapathiraju A. Linear discriminant analysis-a brief tutorial. Institute for Signal and information Processing 18 1998:1-8.

12. Kuhn M. Building Predictive Models in R Using the caret Package. Journal of statistical software. 2008;28(5):1-26.

13. Robin X, Turck N, Hainard A, Tiberti N, Lisacek F, Sanchez J-C, Müller M. pROC: an open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics. 2011;12(1):77.

14. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res. 2014;15(1):1929-58.

15. Sperandei S. Understanding logistic regression analysis. Biochem Med (Zagreb). 2014;24(1):12-8.