Signature Project (DA5030)

Neural network:

<https://www.analyticsvidhya.com/blog/2017/09/creating-visualizing-neural-network-in-r/>

<https://www.r-bloggers.com/2021/04/deep-neural-network-in-r/>

single-cell analysis

<https://journals.biologists.com/dev/article/148/14/dev197962/270926/Single-cell-transcriptomics-of-the-early>

<https://www.nature.com/articles/d41586-018-05882-8>

<https://www.nature.com/articles/s41586-018-0414-6>

**Dataset research:**

**Mouse Nuclear cortex (sc-RNASeq)**

[**https://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression**](https://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression)

*Context:*

Expression levels of 77 proteins were measured in the cerebral cortex of 8 classes of control and Down syndrome mice exposed to context fear to condition, a task used to assess associative learning.

*Content:*

The data set consists of the expression levels of 77 proteins/protein modifications that produced detectable signals in the nuclear fraction of the cortex. There are 38 control mice and 34 trisomic mice (Down syndrome), for a total of 72 mice. In the experiments, 15 measurements were registered of each protein per sample/mouse. Therefore, for control mice, there are 38x15, or 570 measurements, and for trisomic mice, there are 34x15, or 510 measurements. The dataset contains a total of 1080 measurements per protein. Each measurement can be considered as an independent sample/mouse. The eight classes of mice are described based on genotype, behavior, and treatment features.

According to genotype, mice can be controlled or trisomic. According to behavior, some mice have been stimulated to learn (context-shock) and others have not (shock-context) and to assess the effect of the drug memantine in recovering the ability to learn in trisomic mice, some mice have been injected with the drug and others have not.

*Classes:*

1. c-CS-s: control mice, stimulated to learn, injected with saline (9 mice)
2. c-CS-m: control mice, stimulated to learn, injected with memantine (10 mice)
3. c-SC-s: control mice, not stimulated to learn, injected with saline (9 mice)
4. c-SC-m: control mice, not stimulated to learn, injected with memantine (10 mice)
5. t-CS-s: trisomy mice, stimulated to learn, injected with saline (7 mice)
6. t-CS-m: trisomy mice, stimulated to learn, injected with memantine (9 mice)
7. t-SC-s: trisomy mice, not stimulated to learn, injected with saline (9 mice)
8. t-SC-m: trisomy mice, not stimulated to learn, injected with memantine (9 mice)

*Attribute Information:*

[1] Mouse ID

[2:78] Values of expression levels of 77 proteins; the names of proteins are followed by N indicating that they were measured in the nuclear fraction. *For example DYRK1A\_n*

[79] Genotype: control (c) or trisomy (t)

[80] Treatment type: memantine (m) or saline (s)

[81] Behavior: context-shock (CS) or shock-context (SC)

[82] Class: c-CS-s, c-CS-m, c-SC-s, c-SC-m, t-CS-s, t-CS-m, t-SC-s, t-SC-m

The aim is to identify subsets of proteins that are discriminant between the classes.

the Ts65Dn has been used in preclinical evaluations of drugs and small molecules proposed as potential pharmacotherapies for ID in DS.

Ts65Dn mice have shorter life expectancies and show morphological, neurological, and structural abnormalities that parallel those found in patients with DS5,6,7,8,9,10,11,12.

 Ts65Dn mice also display behavioral abnormalities similar to those seen in DS.

Segmentally trisomic Ts65Dn mice provide a postnatal model for Down syndrome

The precise locations of the Chr 16 and Chr 17 breakpoints are 84,351,351 bp and 9,426,822 bp, respectively. The Chr 16 segment contains about two-thirds of the human Chr 21 homologs in the mouse, from the mitochondrial ribosomal protein L39 (*Mrpl39*) gene to the distal telomere. These data were used to generate a PCR genotyping assay for Ts65Dn (Reinholdt et al., 2011), replacing the previous methods of chromosome analysis or qPCR. Northern and Western blotting, enzyme activity assays, and reverse phase protein arrays (RPPA) demonstrate that some but not all genes in the translocation product are expressed at elevated levels in segmentally trisomic animals. RPPA shows a loss of correlation among some brain proteins (Ahmed et al., 2012)

<https://rstudio-pubs-static.s3.amazonaws.com/245066_f7b5962e8ab84594829b84f06ced39b6.html>

<https://www.kaggle.com/datasets/ruslankl/mice-protein-expression/code>

**Project proposal:**

1. What data did you choose? Briefly describe it. Identify the source and where it can be found (*e.g.*, URL)

**Ans:** Data chose: Mice Protein Expression

The data comprises the expression values of **77** proteins and protein changes that generated audible signals in the nuclear fraction of the cortex. There are a total of 72 mice, including 38 control mice and 34 trisomics (Down syndrome) animals. 15 measurements of each protein per sample and mouse were made during the tests. As a result, there are 34x15, or 510 measurements for trisomic mice, and 38x15, or 570 measurements, for control mice. Each protein has a total of **1080** measurements in the dataset.

Dataset found: <https://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression>

Dataset from the study: Ahmed MM, Dhanasekaran AR, Block A, Tong S, Costa ACS, Stasko M, et al. (2015) Protein Dynamics Associated with Failed and Rescued Learning in the Ts65Dn Mouse Model of Down Syndrome. PLoS ONE 10(3): e0119491. [[Web Link]](doi:10.1371/journal.pone.0119491)

1. What are you planning to predict, *i.e.*, what is the target variable? Is it categorical (classification) or continuous (regression)? Is it a data mining task?

**Ans:** The target variable is “class”. The target variable is characterized based on categories like genotype, behavior, and treatment; in combination, the eight groups of mice are outlined. It is a categorical classification. Also, a data mining task.

1. How many rows (examples) and columns (variables) are in your dataset?

**Ans:** There are 1080 rows and 82 columns in the dataset

1. Which algorithms are you planning to use and why? Explain why they are appropriate given the data set, target variable, and features.

**Ans:** For machine learning analysis, we will be performing classification (multi-class classification) as well as clustering (based on behavior and treatment). We will use algorithms such as Naïve Bayes, Rpart, Random Forest, SVM, and logistic Regression for classification because they are all non-parametric and perform well on high-dimensional data. And for clustering, we will perform k-means clustering.

1. Do you expect to do any feature engineering, feature transformations, or other data set shaping? Why and for what purpose?

**Ans:** There is no need for feature engineering or transformation as the variables are independent and numeric. Although the dataset does contain some missing values which can be imputed systematically.

1. How will you evaluate the fit of the algorithms? Which specific evaluation methods will you use and why?

**Ans:** For evaluation, we will perform predictions, estimate accuracy and get confidence intervals. Target distribution using a confusion matrix. We will also draw out the ROC curve to check the model fits.

[**https://towardsdatascience.com/precision-recall-and-f1-score-of-multiclass-classification-learn-in-depth-6c194b217629**](https://towardsdatascience.com/precision-recall-and-f1-score-of-multiclass-classification-learn-in-depth-6c194b217629)

1. Has this analysis been done before? Links? Citations? What will you do differently?

**Ans:** Several analyses have been done on the data set, which you can find here [[Analyses],](https://www.kaggle.com/datasets/ruslankl/mice-protein-expression/code) but most ofthem have used python. Only a handful of them have used R, and out of that majority have conducted binary classification for either of the three classes (genotype, behavior, treatment). So, I plan to perform a multi-class classification with a series of algorithms and use an ensemble model to rate the best algorithm.

[**https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/**](https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/)

[**https://www.kaggle.com/code/nmatorina/random-forest-and-linear-discriminant-analysis/notebook**](https://www.kaggle.com/code/nmatorina/random-forest-and-linear-discriminant-analysis/notebook)

**The data set had many outliers, which can be transformed using log transformation for better predictive power and accuracy. Scaling can be applied to the data, so the data will be normalized. Hence, better predictive power**

[**https://github.com/VErconi/ML\_project/blob/main/ML2020\_Homework.pdf**](https://github.com/VErconi/ML_project/blob/main/ML2020_Homework.pdf)

[**https://github.com/ivanajanickova/GeneExpressionAnalysis/blob/main/Assingment.Rmd**](https://github.com/ivanajanickova/GeneExpressionAnalysis/blob/main/Assingment.Rmd)

[**https://cran.r-project.org/web/packages/FeatureTerminatoR/vignettes/feature\_terminatoR\_howto.html**](https://cran.r-project.org/web/packages/FeatureTerminatoR/vignettes/feature_terminatoR_howto.html)

[**https://cowlet.org/2014/01/12/understanding-data-science-classification-with-neural-networks-in-r.html**](https://cowlet.org/2014/01/12/understanding-data-science-classification-with-neural-networks-in-r.html)

[**https://www.analyticsvidhya.com/blog/2016/03/select-important-variables-boruta-package/**](https://www.analyticsvidhya.com/blog/2016/03/select-important-variables-boruta-package/)

**pca :** [**https://semba-blog.netlify.app/05/13/2020/simple-pca-in-r/**](https://semba-blog.netlify.app/05/13/2020/simple-pca-in-r/)

**dnn:** [**https://www.r-bloggers.com/2021/04/deep-neural-network-in-r/**](https://www.r-bloggers.com/2021/04/deep-neural-network-in-r/)

**h2o:** [**https://h2o.gitbooks.io/h2o-tutorials/content/tutorials/ensembles-stacking/**](https://h2o.gitbooks.io/h2o-tutorials/content/tutorials/ensembles-stacking/)

**reference paper:**

[**https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6349309/**](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6349309/)

[**https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4482027/**](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4482027/)

[**https://bmcmedgenomics.biomedcentral.com/articles/10.1186/s12920-020-00826-6**](https://bmcmedgenomics.biomedcentral.com/articles/10.1186/s12920-020-00826-6)

[**https://watermark.silverchair.com/btm344.pdf?token=AQECAHi208BE49Ooan9kkhW\_Ercy7Dm3ZL\_9Cf3qfKAc485ysgAAAuIwggLeBgkqhkiG9w0BBwagggLPMIICywIBADCCAsQGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMKqhXE2cYECngIQXgAgEQgIIClRxK-ehzXL\_AT94LxAohgh5aoMVRr9WwKy8rRpFb3kvKpdzxsKsLdh01oVxvgNau1-8HlLl8Up\_FE4gPlqlTS7wb9TzDJB8bq4fPux7ErnwEZVMA3IwAF0YfNVN43j4blF78AiGAuSKYjt4VYW442PSuqU9VR-vQtpVHHuh-7GHtYu0hmyHFwKYA8JCYuewenUKB-fniG0OONxxvSMUNwaiMzJaPaoBMqxE8FNPxAyalJBkW3sb0b1JrYtmkjMzoNTQ7m2JRHbNBOfXV\_XP1gXdZ0gDj4NLkcgBHHtqDtnFsG5-whhcXIJRb8Cm3c08ySgYCDFurR-xxZgaTvBv0yJlVJST\_NrLMdOMFLCVSELs9uQTfnIov5h8\_-q4MNUFIHFlPlMxDDh4ukYPOFBVet01r7r2rXM8NcWl7xP9IqPhPDYUP0LWDXcUxzFZcb07o0e7vbd4dF2acOkv6OQQIf-5KEDBTy\_DJS0u2Z5SBKpx4oCXJb8WbO1X\_pavkjMcN4U4j1q0YHxaOGSIGntL2CbCdp7XRvvnbq2ihncL04RKZ-KP4ssiXNHqLaUQOivAbR5m0deo2PgghYoP1Y8XVYnJMWe23OZrV2XSjXE2c3yyNe0ZO6iwE7p4RIU1yfKEFzYA8FkkAr9hwek9Q\_3MkmLf40PI3eoYZxsgJMaqWtuFaGwYzvM9gg77UZb5NQkW4GMwxDnusF9I-sbT882rJUQ3ucd4tjyP5zHsu-k8jjy3n6-Yy-DCneCaCResAdiprRQ18ZqAV12DHsYj3H1j42s\_TyDXyWTlVoib39ssP6Tpgav4y560PQYUmMkx7KqMMZi6aKzNxTNI4T9ebbslKtLQSxlsa37HgcFjk5-bVU50izbKMZx0**](https://watermark.silverchair.com/btm344.pdf?token=AQECAHi208BE49Ooan9kkhW_Ercy7Dm3ZL_9Cf3qfKAc485ysgAAAuIwggLeBgkqhkiG9w0BBwagggLPMIICywIBADCCAsQGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMKqhXE2cYECngIQXgAgEQgIIClRxK-ehzXL_AT94LxAohgh5aoMVRr9WwKy8rRpFb3kvKpdzxsKsLdh01oVxvgNau1-8HlLl8Up_FE4gPlqlTS7wb9TzDJB8bq4fPux7ErnwEZVMA3IwAF0YfNVN43j4blF78AiGAuSKYjt4VYW442PSuqU9VR-vQtpVHHuh-7GHtYu0hmyHFwKYA8JCYuewenUKB-fniG0OONxxvSMUNwaiMzJaPaoBMqxE8FNPxAyalJBkW3sb0b1JrYtmkjMzoNTQ7m2JRHbNBOfXV_XP1gXdZ0gDj4NLkcgBHHtqDtnFsG5-whhcXIJRb8Cm3c08ySgYCDFurR-xxZgaTvBv0yJlVJST_NrLMdOMFLCVSELs9uQTfnIov5h8_-q4MNUFIHFlPlMxDDh4ukYPOFBVet01r7r2rXM8NcWl7xP9IqPhPDYUP0LWDXcUxzFZcb07o0e7vbd4dF2acOkv6OQQIf-5KEDBTy_DJS0u2Z5SBKpx4oCXJb8WbO1X_pavkjMcN4U4j1q0YHxaOGSIGntL2CbCdp7XRvvnbq2ihncL04RKZ-KP4ssiXNHqLaUQOivAbR5m0deo2PgghYoP1Y8XVYnJMWe23OZrV2XSjXE2c3yyNe0ZO6iwE7p4RIU1yfKEFzYA8FkkAr9hwek9Q_3MkmLf40PI3eoYZxsgJMaqWtuFaGwYzvM9gg77UZb5NQkW4GMwxDnusF9I-sbT882rJUQ3ucd4tjyP5zHsu-k8jjy3n6-Yy-DCneCaCResAdiprRQ18ZqAV12DHsYj3H1j42s_TyDXyWTlVoib39ssP6Tpgav4y560PQYUmMkx7KqMMZi6aKzNxTNI4T9ebbslKtLQSxlsa37HgcFjk5-bVU50izbKMZx0)

**Classes in the dataset**

| **Class** | **Type of Mice** | **Type of Experiment** | **Treatment** | **Number of Mice** | **Learning Outcome** |
| --- | --- | --- | --- | --- | --- |
| c−CS−s | Control | Context Shock | Saline | 9 | Normal Learning |
| c−CS−m | Control | Context Shock | Memantine | 10 | Normal Learning |
| t−CS−s | Trisomic | Context Shock | Saline | 7 | Failed Learning |
| t−CS−m | Trisomic | Context Shock | Memantine | 9 | Rescued Learning |

**Summary:**

Essentially, we are doing two analyses i.e., multi-class classification and Feature subset study using PCA and RFE methods for 3 significant learning outcomes.

For multi-class classification, the initial pre-processing involves handling missing values, sub-setting data for classification, and reducing features for improving our analysis. We used 4 models for classification analysis – Naïve Bayes, ANN, Multinomial logistic regression, and randomForest for the stacked learner. For evaluation, we monitored metrics such as accuracy, precision, F1-score, ROC curve, and AUC for all the models and choose the best model that reflected the best performance by considering the above metrics.

For the second half of the project, we tried to extract features (genes) that were potentially responsible for a particular learning outcome and tried to compare our results with the previous studies.

Feature engineering entails reformatting predictor values to make them easier for a model to use effectively. This includes transformations and encodings of the data to best represent their important characteristics.

* Correlation between predictors can be reduced via feature extraction or the removal of some predictors.
* Models that use variance-type measures may benefit from coercing the distribution of some skewed predictors to be symmetric by estimating a transformation.

Feature engineering and data preprocessing can also involve reformatting that may be required by the model. Some models use geometric distance metrics and, consequently, numeric predictors should be centered and scaled so that they are all in the same units. Otherwise, the distance values would be biased by the scale of each column.

One reason why more complex meta-models are often not chosen is that there is a much higher chance that the meta-model may overfit the predictions from the base models.

Note that I set the argument **linear.output** to **FALSE** in order to tell the model that I want to apply the activation function **act.fct** and that I am not doing a regression task. Then I set the activation function to **logistic** (which by the way is the default option) in order to apply the logistic function. As far as the number of **hidden neurons**, I tried some combinations and the one used seemed to perform slightly better than the others (around 1% of accuracy difference in cross-validation score).

**Key points:**

in silico identification of proteins that are significant to the learning process and the immune system and to derive the most accurate model for the classification of mice.

Feature selection: forward feature selection

Another common method for representing multiple features at once is called *feature extraction*. Most of these techniques create new features from the predictors that capture the information in the broader set as a whole. For example, principal component analysis (PCA) tries to extract as much of the original information in the predictor set as possible using a smaller number of features. PCA is a linear extraction method, meaning that each new feature is a linear combination of the original predictors. One nice aspect of PCA is that each of the new features, called the principal components or PCA scores, are uncorrelated with one another. Because of this, PCA can be very effective at reducing the correlation between predictors. Note that PCA is only aware of the predictors; the new PCA features might not be associated with the outcome.

the selected feature subsets not only yield higher accuracy classification results but also are composed of protein responses which are important for the learning and memory process and the immune system

Hsa21 is responsible for nearly 160 proteins-coding genes and five micro-RNAs.

 Over expression of these proteins which include transcription factors, cell surface receptors, protein modifiers, adhesion molecules, RNA splicing factors and components of many biochemical pathways can cause the learning and memory (L/M) deficits. In addition for a person diagnosed with DS, the number of neurons and cellular morphology are not normal in brain regions, such as the cortex, cerebellum and hippocampus

we applied supervised learning methods to protein expression data for 77 proteins (thus a 77dimensional space) taken from the cortex of control and Ts65Dn trisomic mice, with and without memantine treatment and with and without contextual fear conditioning (CFC). We compared our results with previous studies where Self Organizing Map (SOM) was used to pinpoint functional or regulatory similarities among proteins with similar expression profiles.

. Context shock (CS) and shock context (SC) classes were partitioned into memantine or saline injected subclasses yielding four different classes and these four classes were analyzed. Memantine usage improved L/M capability in patients with AD [[22](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6349309/#pone.0210954.ref022)]. Thus, the effects of memantine were assessed for comparison with DS. They showed that more than half of the protein levels changed significantly in the hippocampus. The number of proteins showing important changes in the cortex was smaller

all measurements are normalized with Z-score normalization to prevent proteins with higher values influence on the classification result erroneously

which of the changes seen in control mice are required for successful learning, which of the abnormalities in the Ts65Dn directly contribute to failed learning, and which changes induced by memantine are critical for rescuing successful learning in the Ts65Dn?

If some proteins have values in the range 0–3 and others between 0 and 0.6 (as is the case in our data), then the proteins with higher values would have more influence on the clustering outcome, possibly leading to erroneous results. Thus, all measurements (a matrix where samples/mice measurements are the rows and proteins are the columns) are normalized to the range 0–1, column by column

Dimensionality reduction is decreasing the number of features for identifying the most relevant and important variables. It has the effect of decreasing the computational cost. For dimensionality reduction, feature selection and feature extraction methods can be used. Feature selection chooses a subset of features, while feature extraction generates a new feature set of original features.

 It is the heuristic method that tries to find the optimal feature subset by iteratively selecting features based on the classifier performance. It begins with an empty feature subset and adds one feature at a time for each round. This one feature is taken from the pool of all features that are not in the feature subset and when added it results in best classifier performance

forward feature selection technique is used to obtain the feature subsets for identifying the critical proteins in successful learning, rescued learning and failed learning cases.

To understand which of the protein expression level changes are required for successful learning, all groups of normal mice were inspected in the first case. To determine important proteins in rescued learning, trisomic mice exposed to CFC with and without memantine were analyzed in the second case. The third case found out important protein abnormalities in failed learning by comparing normal and trisomic mice protein expression levels.

 there are 4 common proteins (SOD1, pGSK3B, S6, CaNA) out of 11 proteins which are shown in bold. After literature review, it can be deduced that the selected proteins in successful learning are related to the L/M pathway and the immune responses

Parameters of classifiers are determined based on the grid search hyper-parameter optimization technique which is useful in computational biology problems. With the grid search method, the most suitable parameters for different classification methods are found. In addition, five fold cross validation is applied for preventing overfitting. Together with grid search method, cross validation affects classification accuracy in a positive manner.

PCA is also done for both selected protein subsets and original protein sets for the three cases; successful learning, rescued learning and failed to learn. It is shown that selected protein subsets can better discriminate the class of mice instances when compared with the PCA of the original protein sets for all the indicated cases.

Multiple layers in a deep learning model can learn features from a wide perspective with higher flexibility. Thus, it is logical to obtain good results with DNN. SVM maps data to a feature space and then classifies the data. It explicitly determines the decision boundary directly from the training data. Parameter optimization step is required to build an efficient SVM model. Using grid search method, parameters are selected and SVM with the selected parameters gives higher accuracy results. Accuracy result of random forest is lower than SVM and Deep Neural Network.

The reason of lower accuracy can be the size of the data as random forest generally needs larger number of instances for performing its randomization concept in a good way. Also, decision trees used as base learners in the random forest cannot exactly learn many of soft linear boundaries at the decision surface which can cause lower success than the SVM nonlinear boundaries. Gradient boosted tree is prone to overfitting as it tries to find optimal linear combination of trees about given train data. This tuning stage may be the reason for the lowest accuracy obtained by a gradient-boosted tree.

Imp:

We also observed that Naive Bayes tends to give poor probability estimates when using a large number of features. So feature selection is indeed a good idea here. In addition, it is always a good idea to look into feature selection especially if your feature set is extremely large. If it is done right it can improve the generalization ability of your learning model.

This is a fundamental outcome of the random forest, and it shows, each variable, how important it is in classifying the data. The Mean Decrease Accuracy plot expresses how much accuracy the model losses by excluding each variable. The more the accuracy suffers, the more important the variable is for the successful classification. The variables are presented in descending importance. The mean decrease in the Gini coefficient is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. The higher the value of mean decreased accuracy or mean decreased Gini score, the higher the importance of the variable in the model.

So given some measurements about a forest, you will be able to predict which type of forest a given observation belongs to. Before applying an LDA model, you have to determine which features are relevant to discriminate the data. To do so, you need to use and apply an ANOVA model to each numerical variable. In each of these ANOVA models, the variable to explain (Y) is the numerical feature, and the explicative variable (X) is the categorical feature you want to predict in the LDA model. This will tell you whether the mean of the numerical feature stays the same for each forest type. If it does, it will not give you any information to discriminate the data. Therefore, it'll not be relevant to the model, and you will not use it. However, if the mean of a numerical feature differs depending on the forest type, it will help you discriminate the data, and you'll use it in the LDA model.

Feature Selection: once you have a coordinate space that better describes your data you can select which features are salient. Typically you'd use the largest eigenvalues (EVs) and their corresponding eigenvectors from PCA for your representation. Since larger EVs mean there is more variance in that data direction, you can get more granularity in isolating features. This is a good method to reduce the number of dimensions of your problem

In differential expression studies, it is common to filter out (i.e. feature selection) variables with low variance/IQR

**Stacked learning model:**

* Split the dataset into Train (75%) and Test (25%) dataset.
* Run **3 base models**, such as *Gradient Boost*, *Random Forest,* and *Logistic Regression* using Cross-Validation of 5 Folds
* Stack the 3 base model by applying *Random Forest* and train them. The **X** **features**are the predicted values of the 3 models obtained from the Cross-Validation.
* Compare the **AUC score** of each 3 models and the Stacked one on the Test dataset.

***Stacking*** (sometimes called “stacked generalization”) involves training a new learning algorithm to combine the predictions of several base learners. First, the base learners are trained using the available training data, then a combiner or meta-algorithm, called the *super learner*, is trained to make a final prediction based on the predictions of the base learners. Such stacked ensembles tend to outperform any of the individual base learners (e.g., a single RF or GBM) and have been shown to represent an asymptotically optimal system for learning

We go through cycles that repeatedly builds new models and combines them into an **ensemble** model. We start the cycle by taking an existing model and calculating the errors for each observation in the dataset. We then build a new model to predict these errors. We add predictions from this error-predicting model to the "ensemble of models."

To make a prediction, we add the predictions from all previous models. We can use these predictions to calculate new errors, build the next model, and add it to the ensemble.

There's one piece outside that cycle. We need some base prediction to start the cycle. In practice, the initial predictions can be naive. Even if it's predictions are wildly inaccurate, subsequent additions to the ensemble will address those errors.

**PSEUDOCODE:**