Data Science Project: Epigenetic Classifier for Twin Zygosity

by:

Zexi Liu (813212),
Haoze Xia (1131343),
Atefeh Zamani (1129712),
Ni Zhang (1081143),
Tianyu Zhou (1199306)

Supervisors by:

Prof. Michael Kirley and Dr. Jiadong Mao

Host Organisation:

Twins Research Australia

in the
THE UNIVERSITY OF MELBOURNE

November 2022

THE UNIVERSITY OF MELBOURNE

Abstract

by Zexi Liu (813212),
Haoze Xia (1131343),
Atefeh Zamani (1129712),
Ni Zhang (1081143),
Tianyu Zhou (1199306)

Twins refer to two offspring produced by the same pregnancy, and they can be categorized as dizygotic (DZ) or monozygotic (MZ) twins. In the past, the only possible way to distinguish between MZ and DZ twins was their sex and appearance; However, in one of the most recent studies van Dongen et al. (2021), DNA methylation signature data was introduced to classify MZ or DZ. Based on van Dongen et al. (2021), we validated their proposed classifier and found the results were almost random on some of their training and our external data-sets. Motivated by this, we proposed a more robust classifier model, because it was trained on concatenated data-sets from the different research centres. Further, our method adopted more advanced machine learning techniques and feature selection approaches, yielding an increase of 12.1% in average performance.

Declaration of Authorship

We certify that this report does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of our knowledge and belief it does not contain any material previously published or written by another person where due reference is not made in the text. The report is 7374 words in length (excluding text in images, tables, bibliographies and appendices).

Zexi Liu _____ Date _____

Haoze Xia _____ Date _____

Atefeh Zamani _____ Date _____

Ni Zhang _____ Date _____

Tianyu Zhou _____ Date _____

Contents

List of Tables

Chapter 1

Introduction

1.1 Biological background

Twins refer to two offspring produced by the same pregnancy, and they can be categorized as dizygotic (DZ) or monozygotic (MZ) twins. DZ twins, also known as fraternal twins or nonidentical twins, are two siblings coming from separate eggs, released at the same time from an ovary and are fertilized by separate sperms. On the other hand, if early in development, a single fertilized egg cell, the zygote, is divided into two or more embryos, MZ or identical twins are developed. The process which results in this type of twinning is still elusive and is a long-standing enigma of human developmental biology (van Dongen et al., 2021). Although MZ twinning is a rare event in families, researchers show that the chance of occurrence (prevalence) is similar across the world (3–4 per 1000 births) (Segal, 2017) and is stable with respect to factors such as the mother's age (Hoekstra et al., 2008). Consequently, with no clear genetic or environmental causes, the prevailing hypothesis regarding MZ twining is that it arises at random (Lambert, 2021).

Data-driven ways for classifying if someone is an MZ twin are very useful in medical research. This is because MZ twining is associated with some medical conditions, such as spina-bifida, and some MZ twins may not be aware that they are twins. Possible reasons for the existence of an unknown twin include separation at birth or the loss of twin siblings during a multiple pregnancy (a phenomenon known as vanishing twin syndrome). Hence determining whether a person is a twin based on clinical data may help identify the root of medical conditions related to twining (Segal, 2021).

1.2 Related Work

Our project is mainly motivated by van Dongen et al. (2021), who recently discovered that we can build and classify the zygosity of twins using their DNA methylation information. DNA methylation is a general epigenetic mechanism in the mammalian genome. In mammals, DNA methylation is essential for normal development and is associated with a number of key processes including genomic imprinting and aging (Moore et al., 2013). The methylation data is collected on the CpG sites of human DNA using techniques like Infinium Methylation 450K (Sugden et al., 2020). CpG sites are DNA methylation regions in gene promoters known to regulate gene expression through transcriptional silencing of the corresponding gene (Lim et al., 2019).

van Dongen et al. (2021) demonstrated that, while MZ twins have identical DNA sequences (genomes), they could have different epigenetic features, such as DNA methylation, representing different modifications of gene expression. Moreover, van Dongen et al. (2021) observed that MZ twins have a robust DNA methylation signature in their somatic tissues, and DNA methylation differences in MZ twins are not randomly distributed across the genome. Their research demonstrated that these differences are enriched in certain parts of particular chromosomes and genes van Dongen et al. (2021). Combined, these biological findings justify that it is reasonable to train classifiers from DNA methylation data to classify MZ from DZ twins and other family members (non-twins). Based on their biological findings, van Dongen et al. (2021) trained a regularised logistic regression model using DNA methylation information as training features.

1.3 Novelty and contributions

The analysis of DNA methylation datasets is a challenging task. First, to process the data and to correctly interpret the results both require certain background knowledge about the biological terminology and the biotechnology (e.g. HumanMethylation450 and HumanMethylationEPIC) generating the data. Second, the DNA methylation data are typically ultra-high dimensional, meaning that they have way more features than the sample size. The original datasets we process have around 450,000 covariates. Computational methods have to be scalable enough to handle these datasets. Finally, in machine learning literature, there are a considerable number of methods for building classifiers for high-dimensional data. Benchmarking these methods for our problem and making practical recommendations for biologists are some of the main tasks of our project.

The twin zygosity classifier of van Dongen et al. (2021) is easy to train and scalable to large datasets; however, our numerical analysis implied that their classifier does not have good generalisability when applied to unseen datasets in Appendix A.

In this project, we use a new strategy for building a twin zygosity classifier using DNA methylation data. We first use the random forest to select useful features. Then we apply a variety of algorithms, including logistic regression, naive Bayes, support vector machine, random forest and gradient boosting, to these features to obtain a number of classifiers. Finally, we use ensemble learning algorithms such as stacking and voting to combine the strengths of these individual classifiers.

Our work is innovative in the following respects. First, we use machine learning techniques to automatically select features that are useful for the classification task rather than selecting them manually using biological knowledge. Second, compared with the twin zygosity classifier of van Dongen et al. (2021), we use an ensemble learning approach which tends to overcome the drawback of individual classification algorithms (Mehta and Patnaik, 2021). Our numerical studies in Chapter 3 demonstrated the superior performance of our method. Lastly, to improve the generalisability of our classifier, we utilised multiple datasets rather than just one to train our models and tune their hyperparameters. This is because different DNA methylation datasets are often generated from different tissues (eg blood and skin) using different technologies, so that they tend to have different data distributions (Grundberg et al., 2012). In Chapter 3, we demonstrate that using more datasets in model training increased the robustness of our classification algorithm when applied to unseen datasets.

1.4 Organisation of the report

This project is divided into two major parts. In the first part, the general methodologies of the algorithms applied are introduced along with their principles and necessities. In the second part, the detailed progress of training the new classifier using various machine learning methods is revealed, which consists of data pre-processing, feature selection, modelling training and evaluation. Finally, the experiment results obtained through performance comparison with classifiers provided by van Dongen et al. in 2021 are listed. The validation part of the models introduced by van Dongen et al. regarding new datasets is stated in the appendix.

Chapter 2

Methodology

Inspired by van Dongen et al. (2021), our goal is to build an epigenetic binary classifier for zygosity, using DNA methylation features as input, to achieve higher AUC than the classifiers introduced by van Dongen et al. (2021). In addition to this goal, the client wants to select a relatively small number of features from the data, to make the data collection process easier for medical researchers (i.e biomarkers).

We presents our classification algorithm consist of the following major steps:

- Step1: random forest or logistic regression to select features;
- Step2: train individual classifiers;
- Step3: combining strengths of individual classifiers by stacking or voting.

In this chapter, we describe Steps 1–3 in detail.

2.1 Step 1 – Feature selection

In Step 1 of our classification algorithm, we use either regularised logistic regression or random forests to reduce the original number of features (eg hundreds of thousands) to a user-specified level (hundreds). Since regularised logistic regression and random forests are also used as individual classifiers at Step 2, we describe them in more detail in Section 2.2 and only briefly describe how they are used for feature selection here.

To use regularised logistic regression for feature selection, we first train a lasso logistic regression model with twin zygosity as the response and all DNA methylation features as predictors. See Section 2.2.1 for a detailed introduction to regularised logistic regression

and, in particular, the lasso logistic regression corresponds to the case $\alpha = 1$ in the loss function (2.2). There, by adjusting the weight λ of the lasso penalty term, we can control how many features are selected in the final regularised logistic regression model. We then use these selected features as predictors to train classifiers described at Step 2.

To use random forests for feature selection, we first train a zygosity classifier using all features as predictors. We then rank all predictors according to their variable importance, which is a measure of how predictive a predictor is returned by the random forest algorithm. With this ranking, we simply select a number of most predictive predictors and use them as input to train classifiers at Step 2.

Both regularised logistic regression and random forests are scalable to large sample size and high dimensionality. The features elected by these two methods both lead to reasonable classification performance (see Chapter 3). Hence we recommend using either of them for feature selection at Step 1.

2.2 Step 2 - Individual classifiers

In Step 2, We trained multiple individual classifiers after feature selection.

2.2.1 Regularised logistic regression

Logistic regression is one of the simplest and most commonly used binary classifiers (Hastie et al., 2009), where the response $Y \in \{0, 1\}$ represent the two classes (eg in the zygosity classification, MZ or DZ). Logistic regression models the log-odds using a linear model:

$$\log \frac{P(Y = 1|\boldsymbol{X} = \boldsymbol{x})}{P(Y = 0|\boldsymbol{X} = \boldsymbol{x})} = \boldsymbol{\beta}^T \boldsymbol{x},$$

where $\boldsymbol{x} = (x_0, x_1, \cdots, x_p)^T$ denotes the vector of predictors, including the intercept, and $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^T$ denotes the corresponding regression coefficients. This leads to a model of the probability that the feature vector $\boldsymbol{x}$ comes from class 1:

$$\pi = P(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \frac{\exp(\boldsymbol{\beta}^T \boldsymbol{x})}{1 + \exp(\boldsymbol{\beta}^T \boldsymbol{x})},$$

which leads to the loss function with n pairs of training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1,\ldots,n}$

$$L(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \left(y_i \log(\pi_i) + (1 - y_i) \log(1 - \pi_i)\right). \tag{2.1}$$

When applied to high-dimensional data, standard logistic regression often performs poorly due to the curse of dimensionality (Torang et al., 2019). A common approach for dealing with high dimensionality is to add regularisation terms to the loss function (2.1) of standard logistic regression which penalizes model complexity. In this case, common choices for regularisation terms include L_1 penalty term, leading to lasso regression, or L_2 penalty term, leading to ridge regression. It is well known that the ridge regression is good at dealing with colinearity but it does not perform feature selection, whereas the lasso regression selects features but becomes unstable when strong colinearity is present (Zou and Hastie, 2005).

To combine the strengths of the ridge and the lasso penalties for logistic regression, Zou and Hastie (2005) introduced a general class of regularised logistic regression models, in which the penalty term is a combination of L_1 and L_2 penalty terms. Elastic net logistic regression solves the following minimization problem:

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{N} L(\boldsymbol{\beta}) + \lambda \left[\frac{1}{2}(1-\alpha) \left\| \boldsymbol{\beta} \right\|_2^2 + \alpha \left\| \boldsymbol{\beta} \right\|_1^2 \right] \right\} \tag{2.2}$$

where $\alpha \in [0,1]$ is the hyperparameter the weights for the L_1 penalty $\|\boldsymbol{\beta}\|_1^2$ and the L_2 penalty $\|\boldsymbol{\beta}\|_2^2$, and λ determines the strength of regularisation. When α is close to 1 (resp 0), elastic net regression has a performance similar to lasso (resp ridge) regression. For a fixed $\alpha > 0$, larger value of λ leads to fewer features being selected.

2.2.2 Naive Bayes

Naive Bayes is a class of simple and flexible classification algorithms that is commonly used in natural language processing (Manning et al., 2008). In the binary classification problem, where we have the predictors $\boldsymbol{X} = (X_1, \ldots, X_p)^T$ and response $Y \in \{0, 1\}$, by the Bayes theorem, we have $P(Y = 1|X = \boldsymbol{x}) \propto p(\boldsymbol{x} \mid Y = 1)P(Y = 1)$, where $\propto$ reads 'proportion to' and $p(\boldsymbol{x} \mid Y = 1)$ denotes the probability density of $\boldsymbol{X}$ given that $Y = 1$. The feature vector $\boldsymbol{x}$ is more like to come from class $Y = 1$ if $p(\boldsymbol{x} \mid Y = 1)P(Y = 1) \geq 1/2$. Hence, to solve the classification problem, we only need to estimate the two quantities $p(\boldsymbol{x} \mid Y = 1)$ and $P(Y = 1)$.

Given data $\{\boldsymbol{x}_i, y_i\}_{i=1,\ldots,n}$, where $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$ (here, unlike in Section 2.2.1, we omit the intercept term), we can estimate $P(Y = 1)$ by the proportion of training data from class 1, ie $n^{-1} \sum_{i=1}^{n} \mathbf{1}(y_i = 1)$, where $\mathbf{1}(\cdot)$ denotes the indicator function. The high dimensional density function $p(\boldsymbol{x} \mid Y = 1)$ is difficult to estimate in general. However, Naive Bayes simplifies the estimation problem by assuming that all marginal

distributions $p(x_j \mid Y = 1)$, $j = 1, \ldots, p$, are independent, so that $p(\boldsymbol{x} \mid Y = 1) = \prod_{j=1}^{p} p(x_j \mid Y = 1)$. Under the independence assumption, in order to compute the joint density $p(\boldsymbol{x} \mid Y = 1)$ we only need to compute p marginal (univariate) densities $p(x_j \mid Y = 1)$, which is a significantly easier task. In this project, we used normal distribution to fit the marginal distributions.

Due to the independence assumption, the time complexity of the naive Bayes algorithm for binary classification is $O(p)$, where p is the number of features (Fleizach and Fukushima, 1998). The naive Bayes algorithm is easy to implement and it can be applied to both continuous and discrete data. However, the independence assumption of naive Bayes may be too simplistic in many problems and hence this method can have lower prediction accuracy compared to other more involved classification algorithms.

2.2.3 Support vector machine

The main idea of support vector machines (SVMs) for binary classification is to first project the training feature vectors $\boldsymbol{x}_i$, $i = 1, \cdots, n$ to some higher-dimensional space by some function $\varphi : \mathbb{R}^p \to \mathbb{R}^d$, where $d > p$, and then train a linear classifier in the functional space to separate observations from the two classes. The motivation is that, while the feature vectors $\boldsymbol{x}_i$ from different classes often cannot be perfectly separated by a linear decision boundary in the original feature space, their projections $\varphi(\boldsymbol{x}_i)$ in some appropriately constructed higher-dimensional space can. Hence, by constructing such a higher-dimensional space, we can find a linear decision boundary for the $\varphi(\boldsymbol{x}_i)$'s and then project this decision boundary back into the original feature space, resulting in a nonlinear decision boundary in the original lower-dimensional feature space (James et al., 2013).

SVM uses the so-called kernel trick to ensure that the computation in the higher-dimensional space is scalable. For example, if we use the radial kernel for SVM, the SVM algorithm will construct the higher dimensional space and ensure that, for two feature vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$, the inner product of their projections $\varphi(\boldsymbol{x}_i)$ and $\varphi(\boldsymbol{x}_k)$ in the higher dimensional space can be determined by

$$K(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \exp\left(-\gamma \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2\right),$$

where $\gamma > 0$ is some tuning parameter.

2.2.4 Random forests

To solve the classification problem, the random forest algorithm aggregates a collection of independently built classification trees, where each tree is a binary classifier. A tree classifier can be represented by a tree structure, which is built by iteratively partitioning the data. Initially all training data are in a single parent node, and then the data in the parent node is partitioned into two child nodes. Each child node is then treated as a new parent node and partitioned. This iterative process is repeated until some stopping rule is incurred.

To split a parent node, a feature j is selected and data in the parent node satisfying $x_{ij} > c$ go to one of the child node and those satisfying $x_{ij} \leq c$ go to the other child node, where c is some cutoff value. Here, the feature j and the cutoff are selected such that the resulting child nodes have highest class purity, so that data in the same child node tend to have the same class label. It is well known that classification trees suffer from high variance (Hastie et al., 2009). That is, if the training data are slightly perturbed (eg due to resampling), then the fitted classification tree tend to have a drastically different structure.

Random forests solve the high variability problem of individual classification trees by aggregating them. First we create a large number of bootstrap resamples and train a classification tree from each of the resamples. Then, to classify a new feature vector, we aggregate the classification results of all the individual trees by, for example, counting the majority vote.

Since each classification tree in a random forest is built by splitting according to a single feature a time, random forests are scalable to high-dimensional data. Compared to a single tree model, random forests also output the variable importance of all predictors. When training decision trees, we can compute how much each feature increases the purity of the resulting child nodes. The more a feature increases the purity, the more important the feature is. Random forests produce robust estimate of variable importance by averaging the increase of purity caused by each feature across all individual trees. We make use of the variable importance from the random forest algorithm in our feature selection step described in Section 2.1.

2.2.5 Gradient boosting

Gradient boosting classifier is a classification algorithm that is based on 'boosting weak learners iteratively by shifting focus towards problematic observations that were difficult to predict in previous iterations and performing an ensemble of weak learners' (Dangeti,

2017). This method is a generalization of other boosting methods, such as adaboosting, and applies decision trees in model building.

Compared to random forests, which builds an ensemble of deep independent trees, gradient boosting is based on "an ensemble of shallow trees in sequence with each tree learning and improving on the previous one", Boehmke and Greenwell (2019). As expected, shallow trees are weak predictive models, however, if they are tuned appropriately, they can be "boosted" to produce a powerful model.

2.3 Step 3 - Combining strengths of individual classifiers: stacking and voting

Staking and voting classifiers are sub-classes of the ensemble learning techniques. The structure of these classifiers is based on combining two or more machine learning models to gain higher predictive performance.

Different individual machine learning methods in 2.2 are applied to the same dataset, called base estimators, for both stacking and voting. Both classifiers learn from the base estimators and use their strength to find the final estimator.

The stacking took the base estimators' predictions as inputs and trained another binary classifier to output the final label. We used logistic regression as the final layer for stacking. While for the voting method, the prediction of these base classifiers is combined and the final prediction will be one with the "most vote" or the one with the highest weighted and averaged probability.

Chapter 3

Experiments

Chapter 3 is devoted to introducing total 5 used datasets in Datasets Section 3.1, and details the implementation of our algorithms for these datasets in Implementation Detail Section 3.2. The Implementation Detail Section consists of the Preprocessing Section 3.2.1, in addition to the 3 steps in our proposed algorithms. The preprocessing concatenated all the others 4 datasets for training and leave one for testing, which has made our models more generalizable. At last we presented the empirical performance in the Main Results Section 3.4.

3.1 Data-sets

For this project, the clients provided 5 data-sets to be analysed.

Environmental Risk (E-Risk): This data-set is consisted of whole blood DNA methylation profiles collected at age 18 from 1658 samples, including 430 complete MZ twin pairs, 304 complete DZ twin pairs and 190 participants whose co-twin did not provide blood and did not pass quality control. The genome-wide patterns of DNA methylation were quantified using the Illumina Infinium HumanMethylation450 BeadChip in DNA samples isolated from whole blood.

Australian Mammographic Density Twins and Sisters Study (AMDTSS): This data-set contains the DNA methylation data for 479 women from 130 families including 66 complete MZ twin pairs, 66 complete DZ twin pairs and 215 sisters of twins. The genome-wide patterns of DNA methylation was extracted from dried blood spots. DNA was sodium bisulfite converted using the EZ DNA Methylation-Gold protocol. Epigenome-wide methylation was assessed using the Infinium HumanMethylation450 BeadChip arrays.

Brisbane Systems Genetics Study (BSGS): The BSGS data-set provides DNA methylation data on 614 individuals from 117 families and it is consisted of 135 MZ twins, 223 DZ twins and 256 siblings and parents. Overall design Genomic DNA was extracted from peripheral blood lymphocytes. DNA concentrations were determined by NanoDrop quantification to include 500ng before bisulfite conversion using the EZ-96 DNA Methylation Kit.

E-MTAB This data-set contains data about 648 female individuals, including 240 MZ twins and 408 DZ twins. Compared to the other data-sets that are based on the blood samples, in this data-set DNA methylation was quantified in subcutaneous fat and skin, derived using the Infinium HumanMethylation450 BeadChips.

Epigenetic regulation in the elderly over time (Denmark) The data-set consists of whole blood DNA methylation levels measured in 86 samples from an elderly birth cohort over a ten-year follow-up. Each sample was whole blood drawn in 1997 and 2007. DNA samples isolated from whole blood were quantified using the Illumina HumanMethylation450 BeadChip.

Table 3.1 provides the summary information about these datasets.

TABLE 3.1: The data-sets and the number of MZ, DZ and XZ twins and family members

	Total	MZ Twins	DZ Twins	XZ Twins	Family Members
E-Risk	1658	860	608	190	-
AMDTSS	479	132	132	-	215
BSGS	614	135	223	-	256
Denmark	180	94	86	-	-
E-MTAB	648	240	408	-	-

3.2 Implementation Details

Pre-selected Features from van Dongen et al. (2021)

A subset of 833 epigenome-wide significant CpGs from the meta-analysis studied by van Dongen et al. (2021) was selected to be our training features. In other words, each training instance in our data-sets has 833 predictors and 1 respond label.

3.2.1 Preprocessing

Step 1: The samples that are classified as XZ twins or family members are dropped from our datasets. Hence, E-risk reduces its sample size from 1658 to 1468; AMDTSS

reduces its sample size from 479 to 264; BSGS reduces its sample size from 614 to 358; while Denmark and E-MTAB remain their sample sizes as 180 and 648.

Step 2: 10 samples from the Denmark dataset have inconsistent labels between their ten-year follow-up and are thus considered as invalid data and removed from our dataset.

Step 3: Missing values in BSGS dataset and E-MTAB dataset are replaced by their column means (the mean value of each CPGs site of each dataset).

Step 4 (key step): To produce better generalizable model , we concatenated four datasets as our training data and the left one as our testing data, and this process is repeated 5 times to apply for all our 5 datasets. This has been proven to yield better performance in Table 3.10

More specifically,

TABLE 3.2: The 5 repeats of the training and testing datasets

Repeat	Test Data	Train Data
1	E-MTAB	E-Risk, BSGS, Denmark, AMDTSS
2	AMDTSS	E-Risk, BSGS, Denmark, E-MTAB
3	Denmark	E-Risk, BSGS, AMDTSS, E-MTAB
4	BSGS	E-Risk, AMDTSS, E-MTAB, Denmark
5	E-Risk	BSGS, AMDTSS, E-MTAB, Denmark

Step 5: For each repeat, the training data is further split into two parts: 75 percent as the training and 25 percent as the validation.

3.2.2 Implementation Details for Step 1: Feature Selection

Variable selection using random forest and logistic regression are the two methods applied to each of the five repeats to reduce the number of variables. The SelectFromModel function is utilised, this function automatically selects variables based on corresponding feature importance.

Variable Selection Using Random Forest

As shown in Table 3.3, the number of parameters are reduced from the original 833 to approximately 290 after selecting by random forest.

Variable Selection Using Logistic Regression

The logistic regression as a variable selection method also allows the number of variables to reduce from 833 to around 247, which is shown in Table 3.4 below.

Table 3.3: Number of Parameters After Variable Selection By Random Forest

Test Data	Number of Parameters Selected
E-MTAB	308
AMDTSS	285
Denmark	288
BSGS	299
E-Risk	268
Average	290

Table 3.4: Number of Parameters After Variable Selection By Logistic Regression

Test Data	Number of Parameters Selected
E-MTAB	358
AMDTSS	364
Denmark	326
BSGS	339
E-Risk	348
Average	347

3.2.3 Implementation Details for Step 2: Individual Classifier

Logistic Regression, Naive Bayes, Support Vector Machine, Random Forest and Gradient Boosting are the five models selected for further investigation. Hyperparameter tuning is applied to each of the five models to achieve a better model performance.

Hyperparameter Tuning

Randomized search on hyperparameters RandomizedSearchCV is used to tune the parameters and thus give a better model performance. As RandomizedSearchCV only searches for a fixed number of parameter settings, the computational complexity is decreased than using GridSearchCV.

The following tables show the parameters that are selected to be tuned and the corresponding resulted values after tuning.

Table 3.5: Hyper Parameter Tuning for Logistic Regression

Parameter	Description	Value
penalty	Norm of penalty	”l2”
tolerance	The tolerance for stopping criteria	0.001
solver	Algorithm to use in the optimization problem	”sag”
C	Inverse of regularization strength	30

Table 3.6: Hyper Parameter Tuning for Naive Bayes

Parameter	Description	Value
var smoothing	Portion of largest variance that adds to variances	1e-9

Table 3.7: Hyper Parameter Tuning for Support Vector Machine

Parameter	Description	Value
kernal	kernel type used in the algorithm	"rbf"
gamma	Kernel coefficient for 'rbf'	"scale"
decision_function_shape	One vs One or One vs Rest	"ovr"
degree	Degree of the polynomial kernel function	1
C	Regularization parameter	20

Table 3.8: Hyper Parameter Tuning for Random Forest

Parameter	Description	Value
n_estimators	Number of trees in the forest	500
min_impurity_decrease	A node will be split if this value exceeds	1e-06
max_depth	The Maximum depth of the tree	50
criterion	Function to measure the quality of a split	"gini"

Table 3.9: Hyper Parameter Tuning for Gradient Boosting

Parameter	Description	Value
n_estimators	Number of boosting stages to perform	300
learning_rate	The learning rate shrinks the contribution of each tree	0.5
max_depth	The maximum depth of the individual regression estimators	5

3.2.4 Implementation Details for Step 3: Stacking and Voting

Stacking Classifier

Logistic regression, support vector machine, random forest, gradient boosting and naive bayes of which have their hyperparameters tuned, are stacked as the level-0 models (base-models). Logistic regression with tuned hyperparameters is utilized as the level-1 model (meta-model). The stacking classifier allows all 5 base-models to contribute their strengths to the meta-model and thus achieve a higher AUC than those base-models by themselves.

Voting Classifier

The predicted probabilities of 4 tuned models: logistic regression, support vector machine, random forest and gradient boosting, are aggregated by the voting classifier on a basis of soft voting, in order to resolve the error by any model. The naive bayes classifier is excluded in voting as it has the lowest AUC compared to the other 4 classifiers.

3.3 Evaluation Metric

AUC which stands for Area Under the Curve, is the metric selected to check for model performance. AUC ranges in value from 0 to 1, where 0 means that none of the model

predictions align with the true label and 1 means all predictions are correctly predicted.

3.4 Main Results

The following six classifiers are tested in our datasets. The two NC models are built from the Nature communication paper by van Dongen et al. (2021) and the four stacking and voting model are built in this study. More specifically:

- NC Original: Classifier 13 built from the Nature communication paper. This classifier is trained on the Netherlands Twin Register (NTR) dataset which we do not have access to.
- NC 4 as Training: Classifier 13 but is trained on the combination of 4 datasets instead.
- Stacking RF: Stacking classifier using random forest as variable selection method and is trained on the combination of 4 datasets.
- Stacking LR: Stacking classifier using logistic regression as variable selection method and is trained on the combination of 4 datasets.
- Voting RF: Voting classifier using random forest as variable selection method and is trained on the combination of 4 datasets.
- Voting LR: Voting classifier using logistic regression as variable selection method and is trained on the combination of 4 datasets.

TABLE 3.10: Final Result Table

Testing Data	NC Original	NC 4 as Training	Stacking RF	Stacking LR	Voting RF	Voting LR
E-MTAB	0.522	0.711	0.697	0.684	0.720	0.698
AMDTSS	0.648	0.695	0.701	0.723	0.689	0.717
Denmark	0.544	0.669	0.655	0.638	0.723	0.673
BSGS	0.784	0.808	0.807	0.820	0.808	0.822
E-RISK	0.728	0.732	0.721	0.719	0.734	0.724
Average	0.645	0.723	0.716	0.717	0.735	0.727

The Table 3.10 shows our final result for the six classifiers examined in our study.

Comparing the results of the NC Original classifier and the NC 4 as Training classifier, we can see that the average AUC increased from 0.645 to 0.723. In particular, a huge improvement is illustrated when testing the E-MTAB dataset, which is the only dataset that quantified the DNA methylation level on non-blood tissue. This increase in AUC

demonstrates that more training data can result in an improvement in the generalizability of the model.

Moreover, the Voting RF classifier showcases the highest average AUC compared to all the other five classifiers. With a decrease in the number of variables from 833 to less than 300, this classifier still results in a higher average AUC compared to the NC 4 as Training classifier, implying that the voting classifier has a better performance than the original elastic net penalized logistic regression model.

In conclusion, the voting classifier using random forest to select features is our final model. Compared to the original classifier developed by van Dongen et al. (2021), the number of variables is decreased from 833 to an average of 290, which require a less computational cost. Moreover, the AUC is increased by 14%, from 0.645 to 0.735, which provides higher accuracy in prediction.

Chapter 4

Conclusion and Future Work

In this chapter, we are going to mention some final concluding points about the report along with some suggestions for future work, which might result in some improvements in the introduced classifier.

4.1 Conclusion

Nowadays, it is a well-known fact that, along with appearance and sex, there are more accurate ways to distinguish MZ from DZ twins. Especially, researches show that there is a significant association between Mz twining and DNA methylation, which helps researchers to more effectively examine the mysteries hidden in DNA.

Using the association between DNA methylation and MZ twinning, van Dongen et al. (2021) tried to introduce some classifiers using a penalised regression model to discriminate MZ from DZ.

In the first step of this project, the penalised regression model offered by van Dongen et al. (2021) is validated using various data sets. In Appendix A, it is discovered that the model provides low performance for some of the data sets, due to a lack of generalisability. In particular, the AUC of the test for data extracted from non-blood tissues (i.e., fat and skin) is around 0.52, which is far from the AUC value asserted in van Dongen et al. (2021).

Hence, our research attempts to construct a new data pipeline from a data science perspective and suggests a novel machine learning model with a variable selection approach to increase the accuracy of the model prediction.

Our approach is intended to better model performance and be more generalizable. Our model concatenates all the other datasets for training and leaves out one for testing to improve generalizability. In the meanwhile, our approach will result in smaller variable sizes than the original classifier trained by van Dongen et al. (2021) in Appendix A. Then, in order to choose variables that are more crucial to our model and reduce the number of variables, we used variable selection methods. Finally, to improve model performance, ensemble learning classifiers are trained.

The applied method improves the current methods from the following perspectives.

First, instead of selecting variables based on biological domain knowledge, machine learning techniques are used to automatically select the variables associated with the classifier. The original classifier trained by van Dongen et al. (2021), is based on the 833 features extracted using meta-data analysis. In this project, the applied methods for feature selection are random forest and logistic regression. We have successfully reduced the number of variables to around 300 after applying variable selection.

Secondly, compared to the elastic net penalised regression model, developed by van Dongen et al. (2021), we developed a more advanced machine learning model with higher AUC. The ensemble learning classifiers stacking and voting are chosen to overcome most of the drawbacks of individual classical models and therefore indicate better performance than using classical machine learning models. The voting classifier using the random forest as the variable selection method is our final model. The AUC is increased by 14%, from 0.65 obtained from the original classifier built by van Dongen et al. (2021) to 0.73.

Finally, during pre-processing, we used more data sets for training. Our model leaves one of the datasets out for testing and concatenated all the others for training, which brings diversity in training data and allows an increase in sample sizes. As a result, the patterns and features can be extracted more thoroughly from the combination of different datasets. The average AUC of the classifier built in the Nature communication paper was 0.645 and the average AUC of our model was 0.723, which is an increase of 12.1%. Accordingly, our model predictions are more robust and more generalisable, compared to the previous counterparts.

4.2 Future Work

Along with the improvements gained using the suggested method, the following two points should be covered in future studies to cover the limitations. Firstly, the different instrumentation used to measure DNA methylation data leads to different methods of

normalisation, which may potentially impact the model training as well as the model's generalizability.

Secondly, due to time constraints, the processing of the full dataset (450k CpG) was not completed. The 833 CpG features, used for training the classifier, are extracted from the full dataset using metadata analysis and biological background knowledge and are suggested by van Dongen et al. (2021). In essence, this subjective selection of the features may result in missing some statistically significant CpG sites that were not biologically significant.

Consequently, future work can be suggested to focus on the following aspects. Firstly, the normalisation methods used in our datasets should be further investigated and eventually align with each other. Secondly, we intend to continue to use the original datasets (450k CpG) and the University of Melbourne's HPC (Spartan) to make variable selections based on the existing data pipeline and test the resulting models and compare their model performance with our current classifiers. In this case, the selected variables will be compared with the features in the current model to identify the similarities and differences. A final version of variables will then be provided back to the biologists to examine underlying relations in the CpGs.

In the meantime, our method will decrease variable sizes compared to the original classifier trained by van Dongen et al. (2021) in Appendix A. We then applied variable selection methods to select variables that are more important to our model and thus decrease the number of variables.

Appendix A

Appendix

A.1 Github Repository

https://github.com/MAST90106-90107-2022-Group31/MAST90106-90107-Industry-Project

A.2 Validating the performance of existing classifiers

This section is devoted to the results of applying the classifiers introduced by van Dongen et al. (2021) to the new data-sets provided by the client.

A.2.1 Applied testing method

van Dongen et al. (2021) introduced two classifiers, called classifiers 13 and 14, using elastic net penalized logistic regression. classifier 13 was trained on data from twins only, to classify their zygosity. classifier 14 was trained on data from twins and a small group of family members to distinguish MZ twins from the rest (dizygotic twins and family members). The associated weights for these classifiers are provided by the authors as Supplementary Data 3, Supplementary Data 13, and 14 in the supplementary information. In order to apply these two classifiers to the new data-sets, we filtered the 450K variables to match the exact 251 covariates for classifier 13 and 231 covariates for classifier 14. In the second step, the matched covariates in data and the coefficients are used to compute the elastic net logistic regression responses and, finally, the confusion matrix and AUC scores were calculated.

A.2.2 Main Results

van Dongen et al. (2021) claimed that classifier 13 and 14 have similar prediction results. However, applying these classifiers to data-sets provided by the client leads to some different conclusions. Tables A.1 and Table A.2 demonstrate the performance of the classifiers 13 and 14 for new data-sets. As can be seen in Table A.2, classifier 14 has a better performance when predicting non-twins (family members) than classifier 13. We suspect this might be due to differences in training data-sets, as classifier 14 was trained based on the data-set containing family members while classifier 13 was not.

TABLE A.1: The performance of classifier 13 on different test datasets

Classifier 13	AUC	Correctly predicted MZ twins	Correctly predicted DZ twins	Correctly predicted non-twins
NTR - testing	0.751	0.843	0.469	0.444
BSGS - nature	0.796	0.916	0.450	0.302
E-Risk	0.728	0.684	0.683	NaN
BSGS	0.784	0.904	0.404	0.313
Denmark	0.544	0.596	0.570	NaN
AMDTSS	0.648	0.742	0.545	0.567
E-MTAB	0.522	0.882	0.201	NaN

TABLE A.2: The performance of classifier 14 on different test datasets

Classifier 13	AUC	Correctly predicted MZ twins	Correctly predicted DZ twins	Correctly predicted non-twins
NTR - testing	0.766	0.808	0.572	0.625
BSGS - nature	0.799	0.901	0.468	0.451
E-Risk	0.739	0.621	0.740	NaN
BSGS	0.774	0.637	0.798	0.848
Denmark	0.563	0.755	0.395	NaN
AMDTSS	0.667	0.750	0.530	0.600
E-MTAB	0.492	0.564	0.390	NaN

Moreover, Classifiers 13 and 14 might not be generalizable to the data-sets provided by the client. As can be seen in Table A.3, the average AUC over all test data-sets for both classifiers 14 is around 0.65. This value is close to AUC=0.5, which stands for random classification. Besides, for both classifiers, the data-sets E-MTAB and Denmark have extremely low AUCs, which are around 0.50 and 0.55, respectively. These differences in the performance compared to the ones mentioned in van Dongen et al. (2021) can put extendability of the classifiers to new datasets in doubt.

We have also attempted to validate the correctness of our codes. The BSGS dataset has been tested and reported in van Dongen et al. (2021) and we also have access to it and can run a test on it. As can be seen in Tables A.1 and A.2, the AUCs are very similar,

but the confusion matrix for classifier 14 is different from their results. We will bring further investigation to this issue with the clients.

Classifiers 13 and 14 have low type 1 error but high type 2 error. In the second and third columns of Table A.3, the average proportion of DZ twins correctly predicted is around 0.53, but the average proportion of MZ twins correctly predicted is very high. This pattern may be due to their unbalanced training data. In training data, the number of MZ twins is twice the amount of DZ twins. Therefore, models 13 and 14 may tend to predict more MZ twins in testing.

TABLE A.3: The Average values of the five testing datasets

Classifier 14	AUC	Proportion MZ twins correctly predicted	Proportion DZ twins correctly predicted	Proportion non-twins correctly predicted
Classifier 13	0.645	0.726	0.481	0.440
Classifier 14	0.647	0.665	0.571	0.724

Bibliography

Boehmke, B. and Greenwell, B. (2019). *Hands-on machine learning with R*. Chapman and Hall/CRC.

Dangeti, P. (2017). *Statistics for machine learning*. Packt Publishing Ltd.

Fleizach, C. and Fukushima, S. (1998). A naive bayes classifier on 1998 kdd cup. *Fleizach C. & Fukushima S.(1998). A naive Bayes classifier on*.

Grundberg, E., Small, K. S., Hedman, Å. K., Nica, A. C., Buil, A., Keildson, S., Bell, J. T., Yang, T.-P., Meduri, E., Barrett, A., Nisbett, J., Sekowska, M., Wilk, A. E., Shin, S.-Y., Glass, D., Travers, M. E., Min, J. L., Ring, S. M., Ho, K. M., Thorleifsson, G., Kong, A., Thorsteindottir, U., Ainali, C., Dimas, A. S., Hassanali, N., Ingle, C. E., Knowles, D. A., Krestyaninova, M., Lowe, C. E., Meglio, P. D., Montgomery, S. B., Parts, L., Potter, S. C., Surdulescu, G. L., Tsaprouni, L., Tsoka, S., Bataille, V., Durbin, R., Nestle, F. O., O'Rahilly, S., Soranzo, N., Lindgren, C. M., Zondervan, K. T., Ahmadi, K. R., Schadt, E. E., Stefánsson, K., Smith, G. D., McCarthy, M. I., Deloukas, P., Dermitzakis, E. T., and Spector, T. D. (2012). Mapping cis- and trans-regulatory effects across multiple tissues in twins. *Nature genetics*, 44:1084 – 1089.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hoekstra, C., Willemsen, G., van Beijsterveldt, T. C., Montgomery, G. W., and Boomsma, D. I. (2008). Familial twinning and fertility in dutch mothers of twins. *American Journal of Medical Genetics Part A*, 146(24):3147–3156.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Lambert, J. (2021). All identical twins may share a common set of chemical markers on their dna. https://www.sciencenews.org/article/identical-twin-siblings-common-set-chemical-markers-dna-epigenetics/. Accessed: 2021-09-28.

Lim, W.-J., Kim, K. H., Kim, J.-Y., Jeong, S., and Kim, N. (2019). Identification of dna-methylated cpg islands associated with gene silencing in the adult body tissues of the ogye chicken using rna-seq and reduced representation bisulfite sequencing. *Frontiers in Genetics*, 10.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to information retrieval*. Cambridge university press.

Mehta, S. and Patnaik, K. S. (2021). Improved prediction of software defects using ensemble machine learning techniques. *Neural Comput. Appl.*, 33:10551–10562.

Moore, L. D., Le, T. M., and Fan, G. (2013). Dna methylation and its basic function. *Neuropsychopharmacology*, 38:23–38.

Segal, N. L. (2017). *Twin mythconceptions: False beliefs, fables, and facts about twins*. Academic Press.

Segal, N. L. (2021). Mysteries of monozygosity: Theories and breakthroughs/twin research: Rare case of lost twins; developing a national twin registry; twins' language and gesture delays; dna testing for vanishing twins/media reports: Identical twins discordant for covid vaccination; world's oldest identical twins; olympic athlete stand-in; fraternal twin football players. *Twin Research and Human Genetics*, 24(6):408–412.

Sugden, K., Hannon, E. J., Arseneault, L., Belsky, D. W., Corcoran, D. L., Fisher, H. L., Houts, R. M., Kandaswamy, R., Moffitt, T. E., Poulton, R., Prinz, J. A., Rasmussen, L. J. H., Williams, B. S., Wong, C. C. Y., Mill, J., and Caspi, A. (2020). Patterns of reliability: Assessing the reproducibility and integrity of dna methylation measurement. *Patterns*, 1.

Torang, A., Gupta, P., and Klinke, D. J. (2019). An elastic-net logistic regression approach to generate classifiers and gene signatures for types of immune cells and t helper cell subsets. *BMC bioinformatics*, 20(1):1–15.

van Dongen, J., Gordon, S. D., McRae, A. F., Odintsova, V. V., Mbarek, H., Breeze, C. E., Sugden, K., Lundgren, S., Castillo-Fernandez, J. E., Hannon, E., et al. (2021). Identical twins carry a persistent epigenetic signature of early genome programming. *Nature communications*, 12(1):1–14.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.