Modern statistics in medical and genetic research

Supervisors: Nemes András Botond, Dr. Németh László, Dr. Firneisz Gábor

Somogyi Dalma

Background

The information required for the human body to function is stored in the DNA, built up of nucleotides. These units form genes, that control the biochemical processes in our bodies. In any position, the nucleotide can differ in individuals which is called a single-nucleotide polymorphism (SNP) [10]. The SNPs across the whole DNA constitutes the genotype of the individual that determines its outer traits as well as how its body functions, if it has any diseases, in short, its phenotype [1].

In many cases, SNPs doesn't effect the phenotype individually, but interact with each other, altering the effect they have individually. This phenomenon is called epistasis [2].

The different kinds of SNPs are called alleles. Usually an SNP has two alleles, a dominant a recessive one. With the homologous chromosome pairs, an individual can be of three kind of genotype with respect to a given SNP: homozygous recessive, heterozygous, homozygous dominant [14]. The probability of a recessive allel to emerge is described by minor allele frequency (MAF) [15].

Linkage disequilibrium (LD) is also considerable feature of SNPs. It indicates the association of them, ie. whether their inheritance is correlated. It is often caused by physical closeness on the chromosome. If some level of LD is present, SNPs can't be treated and independent variables [11].

In the early 2000's, the human genome was mapped, producing large-scale datasets containing millions SNPs. Although this provided unprecedented information about genetic variation, much less was known about the biological roles of most genes. This gap motivated the development of genome-wide association studies (GWASs), which investigate whether specific genetic variants are associated with observable traits, such as diseases [4]. The total number of recorded human gene variants to date is over 700 million, however most of them are rare variants. In addition, many large GWASs were already completed for epidemiologically important diseases, such as Type 2 Diabetes Mellitus [8], Hypertension [6]. However, due to compuatational limits, mostly only the strongest associations were found and every new approach can result new discoveries.

In the view of these exceptionally large studies that were already done, in order to end-up with novel results, the approach and/or the outcomes should have a potential for novelty and it is also essential that the most advanced statistical methods are to be applied.

Our study is based on the genetical effects on Gestational Diabetes Mellitus. It is a type of diabetes arising during pregnancy, endangering the fetus if not treated. The standard diagnosis is based on glucose level test at around 24 weeks, that can be sometimes deceptive. By deriving which SNPs and SNP pairs have significant correlation to developing this disease, it could be determined more precisely and sooner [9].

Goal

This semester we were working on the methodology of detecting epistases between SNPs. We aim to find SNP pairs that together have an effect on developing Gestational Diabetes Mellitus while alone they are not significant.

Methods

Epistatic analysis is heavily unexplored area of genomic studies due to the computational capacities. By just testing every pairs among N SNPs for epistatic interactions, we would have to fit $O(N^2)$ models which rapidly explodes given that these studies usually involve millions of SNPs.
That's why we need to come up with statistical methods that decreases the SNP pairs we are scanning without significant data loss.
To work out and test the methodology, we worked on a synthetic dataset with 4000 individuals' 3000 SNPs.
(For this synthetic database to be generated with an R code, genetic variation was simulated by anchoring each block to a "master" SNP with MAF between 0.1 and 0.4, followed by random allele flipping to achieve varying internal correlation levels. The phenotype was modeled by embedding 30 primary causal SNPs and 15 specific epistatic interactions with effect sizes ranging from 0.2 to 0.5. In the end, Gaussian noise was finally added to simulate a more complex genetic dynamics while maintaining known interactions.)
Each SNPs variable can take up a value from $\{0, 1, 2\}$ depending on whether it is homozygous recessive, heterozygous, or homozygous dominant. The target variable is a binary variable indicating the presence of Gestational Diabetes Mellitus.
As per notation, $X \in \{0, 1, 2\}^{4000\times3000}$ denotes the dataset of the SNP variables' samples, every column corresponding to an SNP, and $Y \in \{0, 1\}^{4000}$ the target.
We manually placed 15 significant SNP interactions in it (each within different block pairs, see below) and the goal was to come up with a method that finds as many as possible of them. Note here that even in this much smaller dataset than what is usually used in GWAS, without any statistical consideration, it would require to test for $O(10^6)$ SNP pairs.
The main framework was to group correlated SNPs into blocks, find representative vectors for the blocks, test interactions between these representatives, and then test SNP-level interactions only between those blocks that had significant interactions through their representatives.

LD-based blocks

The most obvious idea was to group the SNPs into blocks based on LD [13].
It can be measured by multiple metrics, the most commonly used one is the correlation coefficient defined as

$$r^2 = \frac{(p_{AB} - p_A p_B)^2}{p_A(1 - p_A)p_B(1 - p_B)}$$

where p_A, p_B, p_{AB} are the frequencies of alleles A, B, and allele pair AB, respectively.
It is easy to see that $0 \leq r^2 \leq 1$. The correlation coefficient indicates the level of joint inheritance of the two alleles, 0 meaning they are inherited independently, and 1 meaning they are almost surely inherited together.
It is thus redundant to test the interaction of SNP pairs with high correlation coefficient, hence we

first grouped SNPs with high correlation coefficients into blocks. The threshold for "high correlation coefficient" was set to $r^2 \geq 0,8$.

Block representatives

The next task was to create a representative vector for every block via different methods.
The columns of the dataset corresponding to the ith block is denoted by X^i, intercept column included.

1st method: Principal Component Analysis (PCA)
A plausible way to find one vector that best describes the effect of the block is PCA.
As a reminder, PCA transforms field of variables so that the first base, called the first principal component explains the biggest part of the sample's variance, the second base, the second PC the biggest part of the remaining variance and so on. One can think of this that it compresses the effects of the database into one vector. Note that this doesn't depend on the target variable, only on the dependent variables:

$$PC_1^j = \text{argmax}_{\|u\|=1}\left(\|X^j u\|_2^2\right)$$

$$PC_k^j = \text{argmax}_{\substack{\|u\|=1 \\ u \perp \{PC_i\}_{i=1}^{k-1}}}\left(\|X^j u\|_2^2\right) \quad (k = 2, ..., \dim(X_1^j))$$

One can either use the first principal component as a representative or the weighted average of the first k principal components if k is the least integer for which the first k PCs describe at least q part of the variance of the block's samples. We set $q = 0,95$ and the weights proportional to the fraction of the variance the PCs describe.
Both ways has its benefits and disadvantages. Using only the first PC may correspond to a lesser part of the variance but it is most strongly describes the effect of the block. Using the weighted average of the first k PC, incorporates more nuances of the effect of the block, but due to summing orthogonal PCs, important effects could be canceled out.

2nd method: ElasticNet [5]
ElasticNet is a logistic (or linear) regression model with a more complex loss function that incorporates both LASSO and ridge penalties.
For the jth block, it fits

$$\text{logit}(\mathbb{P}(Y = 1)) = \beta^j X^j,$$

and the estimate of the coefficients is given by minimizing the loss function below:

$$\hat{\beta^j} = \text{argmin}_{\beta^j}\left\{\|\text{logit}((P)(Y = 1)) - \beta^j X^j\|_2^2 + \alpha(\,\lambda\|\beta^j\|_1 + (1 - \lambda\,)\|\beta^j\|_2^2)\right\}$$

$$1 \geq \alpha \geq 0,\ 1 \geq \lambda \geq 0$$

The two penalties both handles multicollinearity among the variables, but in different ways.
LASSO penalty promotes variable selection, ie. picking one from a highly correlated group, because it forces to discard the redundant ones by minimizing the L^1 norm of the coefficients.
Ridge penalty enhances the coefficient to evenly distribute among correlated variables - and not fluctuate that an overfitted OLS model would do, not able to decide which variable to take from highly

correlated ones - by minimizing the L^2 norm of the coefficients, penalizing more strictly bigger absolute values. We can choose α, the strength of the penalties and also λ, the ratio between LASSO and Ridge. By combining these two penalties ElasticNet could be a powerful tool to handle large datasets with lot of correlated variables.
Understanding the roles of the penalties, for our purposes, LASSO penalty should be the definitive one, so we worked with the standard $\alpha = 1$ and chose $\lambda = 0,9$. We could achieve the best results with this choice out of $\{0.5, 0.6, 0.7, 0.8, 0.9\}$, but more detailed exploration would require infeasible computational resources. However it could be serve for us as a material for future research.
Via ElasticNet, the block representatives were obtained as the fitted linear predictor. If the fitted coefficients were constant 0, then the block was omitted from further analysis. However it is worth to be noted that this doesn't automatically mean it has no significant contribution to the outcome, but can be a result of numerical inaccuracies.

3rd method: Decision tree based models - RandomForest and XGBoost
Decision tree learning is a machine learning technique. For databases consisting of finite discrete variables and a categorization target, it builds a decision tree, starting from a source node, representing the whole dataset. Each path represents a set of variables fixed at certain values, defining a subset of the dataset. The longer the path, the more variables are fixed, creating categories of the samples based on the fixed values. The goal is that at the end of the algorithm the leaves of the tree represent such categories that are (nearly) compatible with the target variable's classification, ie. most of the instances in a tree category fall in the same target category.
The general step for any node is to consider the variables not yet used and split into as many branches as many values that specific variable takes up. The variables based on which we are branching is chosen so that it improves the accuracy of the model the most. The accuracy can be measured by different metrics, the two most commonly used ones are the Gini impurity

$$I_G(p) = \sum_{i=1}^{C} p_i(1 - p_i) = 1 - \sum_{i=1}^{C} p_i^2 \ ,$$

for node p, which is the probability of miscategorizing an instance belonging in p with$(p_1, ..., p_C)$ relative frequencies of classes in p ($C = 2$ for binary categorizations).
or the entropy from an information theoretic approach

$$I_H(p) = \sum_{i=1}^{C} -p_i \log_2 p_i$$

with the same meanings of p and $(p_1, ..., p_C)$.
The splitting of the nodes continue until some reasonable criteria are met, which usually include that the metric doesn't improve significantly anymore, or the tree reached the threshold for depth or leaf number ie. model complexity.
Decision trees can be useful in capturing interactions between variables because they naturally incorporate these effects at the second and deeper levels of the tree (specifically, nth order interactions arise on the nth level).

The model RandomForest [12] trains numerous decision trees based on randomly selected subsets of the data and eventually averages their predictions.

From it, we obtain the representative vector by taking the weighted average of the variables, the weights proportional to the importance of the variables throughout the model fitting.
The importance for parent node p and its c_1, c_2, c_3 children are computed as mean decrease in impurity:

$$\Delta I(t,p) = I(p) - \sum_{i=1}^{k} \frac{|c_i|}{|p|} I(c_i)$$

It is the decrease of impurity upon splitting node p in children nodes $c_1, ..., c_k$ in tree t, where as abuse of notation, by $|n|$ we mean the number of instances belonging to node n, and

$$MDI(I, X_j) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \sum_{\substack{p \in t \text{ branched} \\ \text{based on variable } X_j}} \Delta I(t,p)$$

is the mean decrease in impurity with respect to X_j across all splittings in all trees ($\mathcal{T}$) used during the model fitting, with impurity metric I. So the more X_j is used in splittings and the more those splittings improve the metric, the greater the importance of the variable will be.

The model XGBoost [3] works in a very different way. It improves the model's accuracy step by step. First it builds a decision tree. Then, it repeats to build a decision tree on the gradient of loss of the current model:

$$\frac{dL(Y, \hat{Y})}{d\hat{Y}},$$

then updates the current model by adding this new decision tree to it, weighted by a learning rate η. $\hat{Y}$ is the prediction of the current model and L is a loss function, in binary classifications, the standard logistic loss function:

$$L(Y, \hat{Y}) = -\frac{1}{N} \sum_{i=1}^{N} \left(\hat{Y}_i \log(\mathbb{P}(\hat{Y}_i = 1)) + (1 - \hat{Y}_i) \log(\mathbb{P}(\hat{Y}_i = 0)) \right)$$

It is an efficient way to fit model because it doesn't only takes into account the value of the loss but also its direction.
To avoid too complex models or overfitting, it is also regularized on tree depths, number of leaves, size of leaf values by the appropriate penalties.
To obtain the representative vector, again we weight the variables by importance computed the same way as for RandomForest, using the loss function as metric of impurity.

Block-level interactions

Now that we have a representative vector for each block (B^j for the jth block), first we test for block-level interactions to get a grasp on which block pairs have relevant interactions at all.
We did this in two ways:
First by fitting a logistic regression model on each pair of blocks' representative vectors (B^i is the ith block's) and their interaction term. For the ith and jth block ($i \neq j$), let define their regression matrix:

$$B^{i,j} = \left[1 \vdots B^i \vdots B^j \vdots (B^i \times B^j)\right],$$

where "$\times$" marks a pointwise mutiplication.
Then the logistic regression:

$$\text{logit}(\mathbb{P}(Y=1)) = \beta^{i,j} B^{ij},$$

and the maximum likelihood estimate with respect to the logistic loss:

$$\hat{\beta^{ij}} = \text{argmin}_{\beta^{ij}} \{ \|\text{logit}(\mathbb{P}(Y=1)) - \beta^{ij} B^{ij}\|_2^2 \},$$

Declared the pairs significant with p-values reaching corrected standard $\alpha = 0.05$ level of significance. For correction, here we used a less strict one, Benjamini-Hochberg correction.
Second by fitting an XGBoost model on all the blocks' representatives at once and declaring a block pair significant if their importance is among the top $q\%$. Here for interactions' importance we can use the same MDI formula defined above, with the slight modification that it doesn't considers splits based on one block but "two-level splits" based on block pairs.

SNP-level interactions

After obtaining the block pairs with significant interactions, we now had a much smaller set on SNP pairs on which can perform interaction tests.
For every block pair with significant interactions, again, I tried two different methods:
For one, for every block pair, I very similarly as in the block-level analysis, fitted an XGBoost model on the SNPs in those blocks.
For two, fitted a logistic regression model for every pairs of SNPs and their interaction term, just like with the block representatives. Declared the pairs significant with p-values reaching corrected $\alpha = 0.05$ level of significance. Here we used a stricter, Bonferroni correction, because within blocks there is high LD-correlation.
For that, the number of independent tests performed were needed, which is a bit tricky to compute because of the multiple correlations among the SNPs. A standard procedure in GWASs is to estimate it with

$$\sum_{\substack{(i,j) \text{ significant} \\ \text{block pair}}} Meff_i \cdot Meff_j,$$

where

$$Meff_j = \sum_{i=1}^{M_j} \min(\lambda_i^j, 1),$$

where λ_i^j is the ith eigenvalue of the correlation matrix of the jth block's SNPs [7].
The eigenvectors corresponding to eigenvalues > 1 show linear combinations of SNPs that are correlated, because they explain > 1 variance. Those corresponding to eigenvalues ≤ 1 show SNPs that are not, hence the formula overestimating the independent SNPs within one block. Then, the number of independent interaction tests in a block pair can be overestimated by the product of the independent SNPs in the two blocks.

Results

Xgboost wasn't a successful tool in finding significant block pairs with either representatives, only showing all the real significant pairs when they were accompanied by many other non-significant noise. At $q \sim 99\%$, all the 15 of real significant pairs were detected with many others, and by increasing q to $\sim 99.3\% - 99.7\%$, although noise faded but so did many real significant signals as well.
Logistic regression was more stable choice.
Out of the four methods to select representative vectors for the blocks, the PCA was the most successful. With it, 12 interactions out of the 15 were found, while with ElasticNet, RandomForest and XGBoost 8, 11 and 9, respectively. It must be noted that the remaining 3 the PCA method hasn't found, the other didn't find either. Lowering radically the level of significance in the block-level logistic regressions, the missing 3 ones started to show, but on the expense of many insignificant noise as well.
This, together with the experience with XGBoost, it can be concluded that the signals of the missing 3 block pairs were too weak to detect without redundant noise.
That being said, low level of noise was present in the results of the logistic regression with $\alpha = 0.05$: 3 pairs were found because of correlation to a real significant block pair, so one of their member was an adjacent block to the one in the real pair, eg. with pair (22, 94), pairs (21, 94) and (23, 94) were also found in most cases. Blocks 21, 22, 23 are very small adjacent blocks, so this phenomenon can be explained by both high LD-correlation and/or inaccuracies of LD-block detecting. I tried to workaround this problem by before the Benjamini-Hochberg correction, filtered for these kind of correlated block pairs with with the constraint that they also have to have very close p-values, but the results didn't improve.
Moving on to the SNP-level interactions with the best, PCA method's results. Here the logistic regression and the XGBoost framework both found the real SNP interactions within those blocks that were successfully found in the block-level analysis. I also ran the tests for a manually completed significant block pairs list, but like this, the SNP interactions within those 3 missing block pairs that weren't found on block-level, weren't found here either, again suggesting weak signal of those interactions.
Because of the high LD-correlation within blocks, testing for SNP-level interactions lead to much more significant results than real ones, so I grouped the significant SNP pairs with members within 10 distance of each other and using the means of the members as representative pairs.

References

[1] Molly Przeworski Augustine Kong Alexander I. Young, Stefania Benonisdottir. Deconstructing the sources of genotype-phenotype associations in humans. *Science*, 365(6460):1396–1400, 2019.

[2] Cooper-Knock J. Stamp J. et al Balvert, M. Considerations in the search for epistasis. *Genome Biol*, 25(296), 2024.

[3] Yingjie Guo et al. Gene-based testing of interactions using xgboost in genome-wide association studies. *Front. Cell Dev. Biol.*, 9(16):8011–8013, 2021.

[4] McLean G.R. Franke A. Huang, J. wenty years of genome-wide association studies: Health translation challenges and ai opportunities. *Eur J Hum Genet*, 33:1579–1584, 2025.

[5] Trevor Hastie Hui Zou. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005.

[6] Kamali Z. Xie T. et al. Keaton, J.M. Genome-wide analysis in over 1 million individuals of european ancestry yields improved polygenic risk scores for blood pressure traits. *Nat Genet*, 56:778–791, 2024.

[7] Ji L. Li, J. Adjusting multiple testing in multilocus analyses using the eigenvalues of a correlation matrix. *Heredity*, 95:221–227, 2005.

[8] Zeggini E. McCarthy MI. Genome-wide association studies in type 2 diabetes. *Curr Diab Rep.*, 9(2):164–171, 2009.

[9] Hadarits O Harreiter J Nádasdi Á Kelemen F Bancher-Todesca D Komlósi Z Németh L Rigó J Jr Sziller I Somogyi A Kautzky-Willer A Firneisz G Rosta K, Al-Aissa Z. Association study with 77 snps confirms the robust role for the rs10830963/g of mtnr1b variant and identifies two novel associations in gestational diabetes mellitus development. 2017.

[10] B.S. Shastry. Snps: Impact on gene function and phenotype. *Single Nucleotide Polymorphisms. Methods in Molecular Biology*, 578, 2009.

[11] Montgomery Slatkin. Linkage disequilibrium–understanding the evolutionary past and mapping the medical future. *Nat Rev Genet*, 9(6):477–485, 2008.

[12] Holzinger-E. Dasgupta A. et al. Szymczak, S. A new variable selection method for random forests in genome-wide association studies. *BioData Mining*, 9(7), 2016.

[13] Joseph K. Pickrell Tomaz Berisa. Approximately independent linkage disequilibrium blocks in human populations. *Bioinformatics*, 32(2):283–285, 2016.

[14] Robert R H Anholt Trudy F C Mackay. Gregor mendel's legacy in quantitative genetics. *PLoS Biol*, 20(7), 2022.

[15] Sadeesh A. Srinivasasainagendra V. et al Vejandla, S.C. Calibrating genome wide significance by minor allele frequency across three major populations. *Sci Rep*, 15, 2025.