A non-parametric method for joint association analysis of sequencing and imaging data

Xiaoran Tong

The next generation genome sequencing and neroimaging technology give rise to large, mult-site cohort with growing wealth of next generation sequencing (NGS) data and magnetic resonance imaging (MRI) data, which mandates the corresponding analytical methodology capable of utilizing both type of information to identify predictive biomarkers associated with complex disorders. Such attemp, however, are met with “the curse of dimensionality”, due to the large number of variants in the genome and images. In this work, we tackled the dimensionality issue of 3D cortical surface vertices by training a stacked audoencoder with deep learning algorithm, and generating a compact, abstracted representation of the original surface with the encoder per se. An U statistic with profile similarity based weight term were then adopted to evaluate the joint association of encoded surface and genome data with the phenotype. We showed by simulation that the method maintains the correct type 1 error rate, and achieved a statistical power higher then using either genome or image data alone, or using the original surface, or methods relying on large number of per-variant test. To illustrate our approach, we apply the proposed method to the genomic sequencing and neuroimage data from the Alzheimer’s disease Neuroimaging Initiative (ADNI).

# Introduction

The decade long search of casual variant by genome wide associatio analysis (GWA) hasn’t been satisfying. So far GWA hardly find any single nucleotite variant (SNV) with an large enough effect to act as a stand along necessary cause of any complex diseases. Although a large number of statistically significant common variants were indeed identified by GWA, only a moderate fraction of heritability have been explained by the totality of these findingManolio (2010; Pandey 2010). Despite the setback, human genome is still an intriging source of curiosity owning to its intrisic advantages. When viewed as an exposure, genetic polymophisim is consistant throughout an individual’s life course and all types of organism, saving the complication of study design. Also, as one of the fundermental causes of all biological processes, genomic polymophisim is not suseptable to reverse causality. From a population perspective, the occurance of genetic variation mimic a random assignment of treatment in an quasi-experiment, which in turn can be exploited to infer non-genetic effect through an intrumental variable approach, such as Mandilian Randomization Lawlor et al. (2008; G. D. Smith and Ebrahim 2003). These features keeps genomic analysis a promising tool for casual inference.

## Analysis of Rare Genomic Variants

The “rare variant, common disease (RVCD)” hypothesis aims to to explain of the “missing heritability” which GWA failed to capture. RVCD states the gap could be attributed to rare variants of moderate to large effect not covered by GWA Cirulli and Goldstein (2010). The Next Generation Sequencing (NGS) projects, growing in both number and scale over the last decade, offered numerous data sources for the analysis of rare variants. However, the stockpiling data also raise a number of methodological challenges. For one, the variants in a NGS profile is much denser than a GWA profile, which poses intense computation and multiple testing should the traditional per-variant based screening procedures is applied. Also, as the name suggested, the newly detected rare variants came with their minor allele frequencies (MAF) close to 0. As a consequence, the lack of heterogeneity in genotype threatens the statistical power for studies of moderate and small sample size. So far the most popular remedy is signal aggregation, that is, instead of screening the whole profile one variant after another, we first group the variants accroding to certain criteria, and subsequently all the variants in one group are tested together as an unit, during which the signal of the members are aggregated in certain way. The aggregation can be achieved by either collapsing the grouped variants into a single variant Madsen (2009) before the statistical test, or by testing all the variants together with a multivariant approach Wei, Elston, and Lu (2015; Wu et al. 2011). The aggregation can also be done after a per-variant screening, by summarizing the test statistics (e.g. p-values) of group members into one statistics Dai, Leeder, and Cui (2014; Purcell et al. 2007). Grouping and aggregation drastically reduce the number of hypothesis to be tested, and a grouped testing unit have a much improved heterogeniety over any of its member variant. A difficulty comes alongside this approach though, is the choice of grouping criteria. The most common criteria refer prior knowledge of biological function, resulting in gene or pathway based grouping. One could also group the variants by every few kb[?] or by a threshold of linkage disequilibrium (LD) Purcell et al. (2007). Since the focus of this study is hypothesis testing procedure, for now, we adopt a gene based grouping scheme for both simulation and real data analysis.

## Incorporating Imaging Data

An important factor contributing to the unsatisfactory performance of GWA is the genetic effect being intrinsically weak over a complex disease, due to the large “black box” between the upstream genomic variants and the health outcome at the downstream far end. It is desirable to probe the “black box” by incorporating other intermidiate biological profiles. The added information should increase the chance of detecting strong association, especially when the new profiles is in the casual pathway from the genomic profile to the disease. Our method borrows additional information from neuroimaging data, knowning that the brain structure is a powerful predictor of neurological disorder, and its association with the genomic profile can be captured with proper statistics. Much like the genomic analyis, similar procedures can be applied to image profiles given if proper definition of a variant and its value is given. Taking the structured MRI as an example, it is natural to view a voxel in the pile of slices as a variant, and the normalized brightness of that voxel as its value. Following the definition of image variant and value, as an analogy of GWA, a per-voxel screening procedure can then be applied to the entire profile to detect signifiant loci in the brain, which is usually called voxel-wise analysis (VWA) Ashburner and Friston (2000; S. M. Smith et al. 2006; J. Baron et al. 2001; Chételat et al. 2005). For our study though, the neuroimaging profiles are first processed by [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu), a freely distributed neuroimaging processing pipeline. Thus, instead of the orignal structure MRI slices, the 3D cortex reconstructed from these slices is treated as our image profile, and corespondingly, the image units are vertices spanning the cortical surface instead of voxels in the original MRI slices. The software package [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu)also came with its own package for vertex-wise analysis (also abbreviated to “VWA”), which is essentially applying generalized mixed linear model for each vertex Bernal-Rusiel et al. (2013; Salat et al. 2004). The per-variant screening of image profile may work better than the same procedure applied to genomic profile since the continuous value of a image variant (e.g., brightness of a voxel or thickness at a vertex) do not suffer the low variability issue like a rare genomic variant does. However, the multiple testing might be severer than the genomic profile because the variants in a image based profile are densely placed in the 3D space, so the neighboering variants cover tightly connected cerebral tissue, which means their values are highly coorelated. Thus, the grouping and aggregation strategy is also practical for any analysis involving image profiles, which is also adapted by our method. As an analogy of gene based grouping and signal aggregation, it is most nature to segregate the cortex into 68 well recognized functional anatomial regions (34 for each hemisphere) and treat each of them as an analytical unit. In this study, we will compare the power performance of aggregated unit with the per-variant based VWA.

## Dimension reduction using Deep Artifical Neural Network

The typical testing unit of the image profile is also high dimensional, oftentimes even higher then most of the genomic testing units. In our current study data, after the cortex profile being partitioned into 68 functional anatomy regions by [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu), the chance of a randomly picked cortex region holding more variants than a randomly picked gene is as high as 96.8%. Although the multivariate algorithms that works with genomic testing unit can be applied to the cortex with little to no rework, it would be desirable to first reduce the dimensionality of a image based test unit before the analysis. One popular trend is to ultilize deep artificial neural network to extract high order features from the raw profile, which not only lowers the dimensionality but also raises the signal to noise ratio Hinton and Salakhutdinov (2006). An important potent of deep network lies in its capability to accumulatively encrypt incoming knowledge with unsupervised training technique. In other words, as long as the future collection of data profile follows protocols compatible to the current, the existing deep network can be incrementally refined to achieve feature extraction that is more informative and more concise. For now, our method included a stacked autoencoder (SA) to process the cortex profile, which is a ralatively light weighted deep network whose feature extraction capability is comparable to the more powerful but much slower deep belief network Vincent et al. (2010; Glorot and Bengio 2010). The construction and calibration of the stacked autoencoder is detailed in the method section and the appendix. If the feature abstraction of the cortex profile is proven to be helpful, the deep neural network could also be used for the futuristic genomic analysis, because the dimensionality of genomic profile is growing alongside with the advancing genotyping technology and sample size, which could benifit from high order feature extraction as well.

## Association Analysis of multiple high-dim Component

The introduction of medical image also complicated the constitution of association. In most cases, an investigator does not know the effect composition in advance, that is, the variation of phenotypes can be attributed to either genomic polymorphism or cortical vertex alone, or both, either with or without some unknown type of interaction, mediation or even feedback loops among genomic, cortex and phenotype profiles. Thus, the method in mind must be sensitive to the association but at the same time robust enough to maintain statistical power when the putative mode is unavoidably mis-representing the reality. Also, the value of genomic and cortical variants could come from distinct, non-normal distinct distriubtions, while the phenotype profile could also be multivariate (e.g. disease diagnosis with additional demographics and known risk factors), with each element following unknown distribution. Taking these uncertainty into consideration, the test statistic should also be versatile enough to counter an admixture of possibally skewed, non-normally distributed data component. More ever, even with the grouping and aggregation technique, the method has to be reasonablly fast and powerful, because the combination of two or more profile types can be a huge number (in our case, there are approximately 21,000 genes and 68 cortex regions), which is a computation and multiple testing challange. , and the number with the high dimensional profiles comprised of up to tens of thousands genomic polymophisms or vertices (before being replaced with high order features).

In general, we are facing an increasingly denser genomic profile with a growing number of rare variants, and the oppertunity to incroporate additional high dimensional biological profiles in to the genomic analysis. These large data however, while offering new possiblities for the exploration of missing heritability, also poses serious methodology challanges in terms of lower power and high dimensionality. We propose use of a similarity U statistics for the association analysis involving genomic and cortex variants, and compare the power performance with the the per-variants based vertex-wise analysis. In addition, we will train an stacked autoencoder with the cortex profile of all available samples, and use it to extract high order features from the profile, after which another U statistical test will be derived by replacing the cortex vertices with the extracted features, and be compared with the original statistic.

# Material

Next generation sequencing (NGS) and magnetic resonance imaging (MRI) data were obtained from Alzheimer’s Disease Neuroimaging Initiative (ADNI). A totol of 808 subjects at the screening and baseline of ADNI1 and ADNI2 study have both types of profiles available, alongside with disease diagnosis, demographics, and the genotype of APOE 4.

The image preprocessing is done by the [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu)suite freely distributed online (<http://surfer.nmr.mgh.harvard.edu>) Fischl (2012). A variant is then defined as a vertex in the reconstructed surface, and its value being the thickness of cerabral gray matter at that vertex.

The structure MRI is data first went throught a series of preprocessing including special registration, skull stripping, cortical/subcortical segmentation, white/gray matter segregation, vexel intensity normalization, reconstruction of cortical surface, surface registration, and surface paceration. The entire pipline is implemented by [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu)- a neuroimage analysis package developed by Fisher and Dale et.al. Fischl (2012), and currently maintained by *the Laboratory for Computational Neuroimaging (LCN)* at *the Athinoula A. Martinos Center for Biomedical Imaging*. [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu)is freely distributed online (<http://surfer.nmr.mgh.harvard.edu>). For each cerebral hemisphere, the cortical surface reconstructs from the MRI slices is spaned by vertices in the 3D space, connected by triangles. Each vertex is then treated as a variant, with a number of geometrical attribute calculated during the surface reconstruction attatched to it, such as the coordinate of the vertex, the gray matter thickness, average curvature, local area and volume around its vicinity. Of all these attributes bounded with each vertex, that is, a image variant, currently we took the gray matter thickness as its value. The last preprocessing step of [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu)– surface paceration, would divide the entire cortical surface into 68 anatomical regions of interest (ROI), 34 for each hemisphere. For real data analysis, the 68 ROI are treated as our testing units, for simulation study, small ovals of vertices (mean diametter=28mm) were randomly picked from either hemisphere surface for each iteration.

Rigorous quality control had been done by ADNI during variant calling process, thus the WGS data from ADNI do not require intensive preprocessing. For genomic profile, the testing unit for both real data and simulation study are based on gene. The chromosome location of known genes were queried from the table of genomic features of human reference genome assemble version 38, maintained by Genome Reference Consortium (GRCh38). An extra 5k flanking basepairs were attached to both ends of a gene when searching for vairants in a testing unit. Despit the added flanking region, some unit contains no genomic variant, and they were excluded from further opertations.

# Method

## Generalized Multivariate Similarity U Statistic

The goal of our method is to jointly test the possible association among the genomic, cortex and the phenotype profiles. The mothod in mind must be robust, versatile and fast in order to minimize the power lost due to model mis-specification, to avoid the bias due to strong distribution assumptions, and to limit the time need for screening a large number of high dimensional combinations. In this study we implement the Generalized Multivariate Similarity U statistic (GMSU) postulated by Changshuai et. al. Wei, Elston, and Lu (2015). GMSU is computationally efficient, and highly flexible since no distribution assumption will be imposed on genomic and cortex profiles, together with a rank based normalization procedure, the U statistic is also invulnerable to multidimensional phenotype whose elements come from a mixture of unknown distributions. To derive the generalized similarity U statistic, three kernel functions are chosen for each of the profiles accordingly. A kernel function measure the similarity between a pair of samples with respect to one of the profiles. Depending on the charisteristics of that profile, the exact form of kernal function can be flexible, as long as it is symmetric and has finite second moment. Thus, the similarity measurement is a valid U kernal function if and are satisfied. For the current stuty, the kernel functions are chosen according to common practices.

For genomic variants taking values from discrate minor allele count , the common choice of similarity mesurement is the identical by state (IBS) kernel function [eq:wSG]

where and is the value of th. variant in the testing unit (e.g. a gene) taken from the th. and th. samples, respectively, and is the dimensionality of the testing unit (e.g. number of polymorphism in a gene). is the weight assigned to the variant according to *a prior* hypothesis, an example is the minor allele frequency (MAF) based which gives more emphasize on rare variants. Without prior knowledge though, the IBS kernal is simplifed to by setting .

For cortex profiles whose variants (the vertices) take continuous values within , we use the euclidian distance based kernel function [eq:wSV]

to measure the simiarity between sample and , which is also called a Gaussian kernel. Here and are values of the th. vertex in the cortex testing unit of the th. and th. sample, respectively, and denote the number of vertices in the testing unit. The vertices in the cortex profile can also be weighted by vector , but for now we have no prior knowledge of the relative importance of the vertices, the Gaussian kernal function is simplifed to .

Lastly, for a multivariate phenotype profile whose elements may be drawn from a variety of unknown distributions, we first normalize its elements with the rank normal quantile function

where is the th. element of the phenotype profile, is the dimensionality of the phenotype (i.e. number of elements), and is the number of samples. Doing so not only corrects skewed elements, but also bypass the complication of admixed distribution types introduced by a multivariate phenotype. As a result, the pairwise similarity with regard to phenotype can also be measured by a Gaussian kernel function

where is the values of the th. element of the normalized phenotype profile of the th. sample, again with weight denoting the ralative importance of every phenotype element. For a phenotype with only one dimension, that is, , the above measurement simplifies to .

All three kernel functions must be centralized, which is done by substracting the function value at each pair with the two marginal mean of all pairs involving and , respectively, then adding the overall mean of all pairs to it Wei, Elston, and Lu (2015). Taking the kernal function of genomic profile as an example, the centralized similarity measurement is

where is number of samples.

Finally, the generalized multivariate similarity U statistics is the mean product of three similarity measurement of all pairs except the self-pairs, which is

Under the null hypothesis that no correlation exists among all three profiles, the mean product of all pairs of similarity measurement should be since all three kernel functions are centralized. If the value of significantly deviates from , the null hypothesis should be rejected and some association among the three profiles is detected. Under the null, follows a mixture of distrubtion, the value can be calculated using Davis method Wei, Elston, and Lu (2015).

By dropping the kernal function of cortex vertices , the simplified similarity U statistic

tests the more specific null hypothesis that no association exists between the phenotype and genomic profiles. Likewise, by dropping the kernel function ,

test the null hypothesis that no association exists between the phenotype and the cortex profiles.

## Stacked Autoencoder

The stacked autoencoder is an artificial neural network mimicking sentimental visual processing, its purpose is to abstract high order features from the raw image profile. The high order feature not only has lower dimensionality, but is also more relevent to decision making. Taking our data as an example, being able to see the approximate location and size of the laceration sites in the cortex, is far more important than knowning the exact thickness, curvature and coordinates of every vertex in the raw profile. Thus, besides dimension reduction, we also anticipate a power boost for any similarity U statistics involving cortex profile if the raw vertex data is replaced with abstracted feature.

An SA is formed by layers of autoencoders stacking on top of each other, hence the name “Stacked”. An autoencoder layer performs a linear recombination of the input elements, followed by an element-wise non-linear transformation. Usually, we make sure the output has a lower dimensionality then the input to ensure feature abstraction and dimension reduction. The autoencoder at the th. layer of the stack is written as:

where is the layer output and is the layer input, which is also the output of the autoencoder from down below, that is, the th. layer in the stack. The cross product between the input vector and the weight matrix followed by the addition of the offset achieves the linear recombination of input elements. The superscript and denote the dimensionality of data and structure parameters of the autoencoder layer. As mentioned before, to ensure feature abstraction and dimension reduction actually happens, is made smaller than . For our method, a autoencoder layer always halve the dimension of its input, that is, = . Lastly, the inverse logit is chosen for the elementwise non-linear transformation, thus

where is the linear recombination of the input ; the super script denotes its dimensionality and indexes its elements. The “S” shaped inverse logit curve resembles the biological activation of the th. neuron in the th. layer of visual cortex when the weighted sum of simulations from all neurons in the previous layer, , exceeds a threshold. The weight is taken from the th. row vector of , and the threshold is the negation of th. elements in the offest vector .

An SA of layers, of dimensional raw input and dimensional output , is assembled by recursively taking the output of the lower autoencoder layer as the input of the layer above, and ensuring the dimensionality of the output at the top is .

where is the dimensional raw input of one individual, which is viewed as the output of non-existing th. autoencoder, with . Reading from bottom to top, the SA gradually abstracts higher order features from the dimensional raw input , until the dimensionality of the output is as low as .

The SA thus constructed is worthless without calibration. One must find the set of structure parameters that best represents the body of knowledge regarding the data, which, in our case, is the knowlege of human cortex. Only then the SA is truely capable of abstracting meaningful features out of the raw input instead of haphazardly reducing it into a small but irrelevent output (e.g. a vector of random numbers). The “goodness of abstraction” can be inferred from the losted detail disagreement between the raw input, , and its mirrored self, , which is the input reconstructed from , the abstracted high order features. The rationale is that, the disagreement between and measures how badly the restoration resemble the true original, which indirectly tells us how poorly the encoder had performed, because, a superior abstraction should be less likely to obstruct the recovery effort. Thus, the set of parameter that minimize the difference between the orignal and the reconstructed will be considered the optimal configuration of the SA. The calibration guilded by such criteria is called unsupervised training, or unsupervised machine learning. The term “unsupervised” states the fact that no external knowledge other than the raw input is needed. Instead of tuning the paramters to appeal a certain problem (e.g. logistic regression aims to maximize the classification accuray), unsuersvied learning encourage the SA to turn itself into an encrypted knowlege of the data of interest. Not requiring labeled data is the greatest strength of unsurpervised learning (e.g. logistic regression requires not just , but pairs of to fit regression coefficients), which in turn allows a much larger pool of samples to contribute to the calibration, and, as new samples keep popping up, the SA can be continuously refined, mimicing a sentient being’s capabilty of learning new knowlege. In particular to our method, unsupervised learning ensure all 806 samples could contribute their cortex profiles to construct the SA, even if 427 of them cannot enter the joint U statistical analysis due to uncertainties in diagnosis.

The new issue on the table is how to reconstruct the input, that is, a decoder counterpart of the stacked autoencoder is needed. The most nature way to build a decoder is to mirror the encoder structure, thus the decoder will also be a stack of layers, each layer also performs linear recombination of its input, followed by a non-linear, element-wise transformation, but, the dimensionality change is in exactly reverse order of the SA. By mirrowing the th. encoder in the SA, the th. layer in the decoder stack is

With the above layer definition, the decoder stack can be assembled in the same way the encoder stack was assembled. Continue with the layered example SA in [eq:ES], its decoder counterpart is

Reading from bottom to top, the decoder gradually adds details back to the abstracted feature , and eventually produce a retored state of the raw input on its top, denoted by . The restoration process is reflected, and driven by the dimenality change from to , which is in exact reversed order of the SA. Now with both encoder and decoder stacks ready, the complete cycle of encoding and reconstruction is done by treating the top output of the SA ([eq:ES]), that is, the abstracted code , as the lowest input of the decoder stack. The combined the structure is

In addition to structure mirroring, a common strategy to train an stacked autoencoder is to constrain the weight matrix in a decoder layer to be the transpose of its encoder counterpart, that is, by forcing , the th. decoder layer become

Our method adopts this strategy, which not only halve the number of parameter to be tuned but also follow the common sense that encoding and decoding are essentially symmetric concept. More importantly, the constraint encourages the optimization to create an optimal SA, not an inferior SA coupled with a superior decoder stack on its top. Afterall, our best interest is the high order features , not the reconstructed input .

The next thing to do is to measure the total disagreement between original profiles and reconstructed profiles for all the samples, which is called the reconstruction loss . For now, the most popular form is cross entrophy

The two matrices and store the original and the reconstructed profiles of all samples, respectively, with each individual sample indexed by , and the elements in each sample profile indexed by . One could view the restoration of from as an array of binary classification problems, with the true probabilities being , the predicted probability being . The reconstruction loss closely resembles the deviance of a logistic regression analysis which measures of how badly the fitted model reflects the observed reality. With the reconstruction loss defined, the calibration of SA become a numerical optimization problem

With the constraint ([eq:CW]) on the weight matrices in the decoder stack, the set of parameters to be tuned is

whose size is , which is parameters less then the non-constrained decoder. The optimization procedure is is done by gradient guided iterative algorithm, which is covered in the appendix section.

The optimization is computational intense, because the number of parameters is usually large. When the number of layers in the encoder – decoders stacks is also large, the computation use to be inhibitively hard, because with the same number of allowed parameters, the complexity of the function represented by a network grows exponentially with the number of layers, that is, a deeper SA has more local minimum in the parameter space for the reconstruction loss to fall into. Yet, complex function also stands for high flexibility, which makes deeper network enormously intrigging, since a exponentially richer funtion space means a much better chance to find a network that could further reduce and at the same time produce even more compact abstraction. Deep artificial neuro networks have revived its popularity in recent years, thanks to the break through in its training procedure, which is now popularly dubbed “deep learning”. For our method, we implement the layer-wise greedy pre-training procedure Bengio et al. (2007; Vincent et al. 2010). The idea is to first train each encoder layer separately, then fine tune of the entire structure afterwards. To perform the layer-wise pre-training, the output of th. autoencoder is not sent to the th. autoencoder like ([eq:ED]) does, but instead is redirected to its decoder counterpart immediately to produce an intermidiate reconstruction ,

where . The th. encoder–decoder tuple can be trained by minimizing the local reconstruction loss , which is fairly easy thanks to the small number of parameters in . A total of such tuples is formed and trained separately. What the greedy layer-wise pre-training has done is non-randomly initialize the entire structure to a state closer to optimum. After pre-training, all the encoders and the decoders are wired together like [eq:ED] and fine tuned together. The comprehensive fine tuning will reach convergence much faster and less likely to fall into a poor local minimum then a direct training scheme without the pre-training.

## Implementation

The method we propose will be based on the joint GMSU statistic formulated with three kernel functions corresponding to the three profiles. To test the rubostness and versatility of the proposed method, we simulated continuous and dichotomous phenotypes from the effect of purely genomic, purely vertex and mixed basis. Under these scenarios, the statistical power of is compared with simplified and whose null hypothesis is more specific but supposedly less flexible.

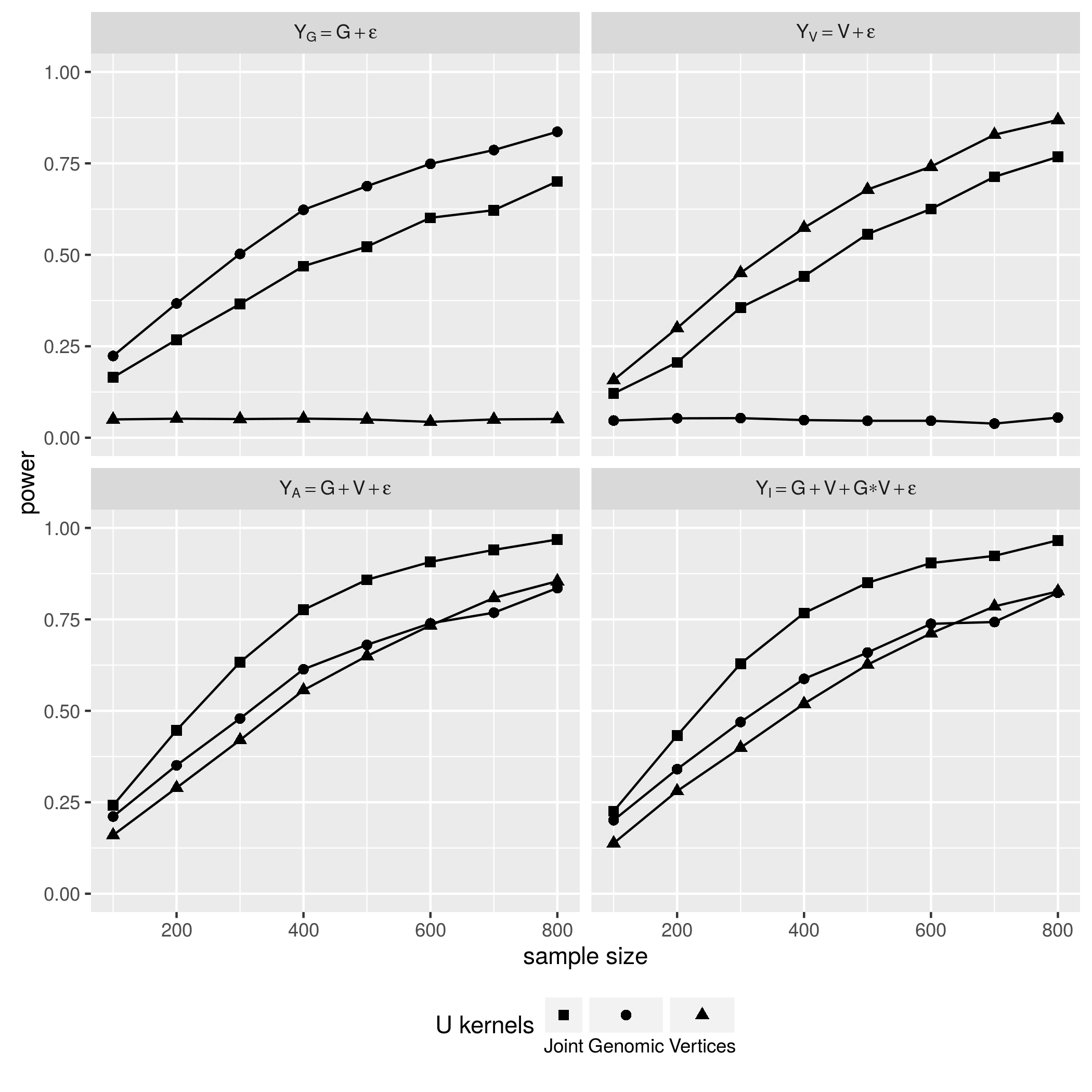
The method also suggests replacing the vertices in the cortex profile with high order features abstracted from them through the training of a stacked autoencoder (SA). In every iteration of the aforementioned simulation scenarios, all 806 cortex sample profiles are utilized to train a 4 layered SA, which is used to replace the original profile with the abstracted features whose dimensionality is 16 times smaller. The newly derived U statistics are compared with the one relying on the original vertices.

Lastly, we pick out 327 individual whose real diagnosis status is certain (either healthy control or Alzheimer’s disease/Dementia case) and apply the proposed method to screen for associations among diagnosis, genes and cortex regions.

# Simulation

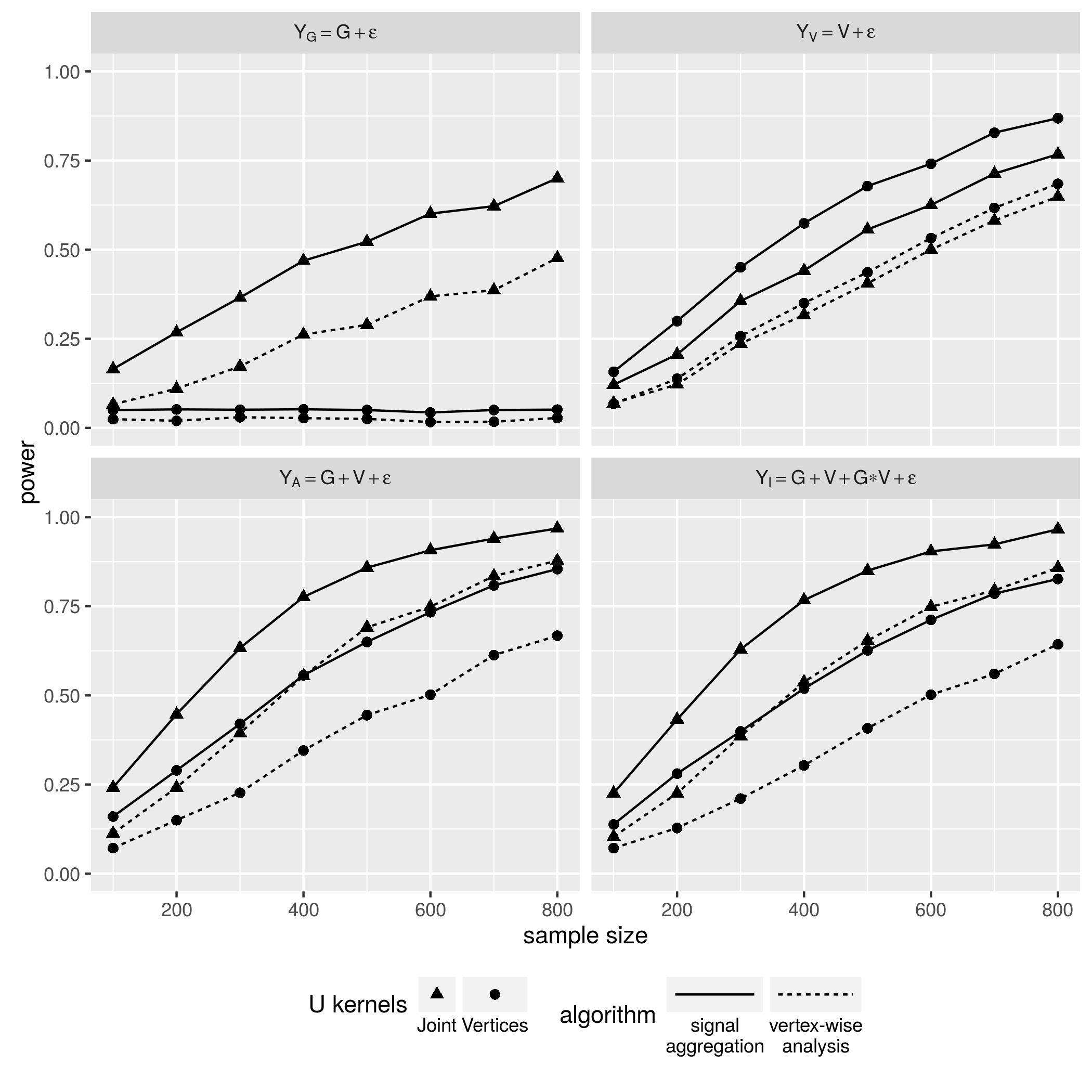
The simulations are based on the real NGS and MRI data of 806 ADNI participants with both profiles. Each iteration run choose a pair of genomic and cortical testing units. The genomic testing unit is a gene with 5kp upstream and downstream flanking window. As for the cortex, a testing unit is a region of 512 vertices randomly picked from the entire cortical surface, which is is roughly an oval of 2.8mm in diameter. The genomic effect and vertex effect are simulated by assigning values drawn from standard normal distribution to a certain percentage of the variants randomly selected from a testing unit (e.g. polymophisms in a gene or vertices in an oval cortical region). The purely genomic and vertex based response are then generated as the product of the testing unit with the simulated effect. An additive and an interactive response are also created by adding up the two basic responses, with and without an additional element-wise product of the two. Lastly, we assign some noise to the generated responses. For now we will focuse on continous respones, the simulation study for dichotomous responses will be covered later.

**Robustness of the Joint U**  
The first set of study aims to test the robustness of the joint U statistics under very likely circumstances of model mis-specification. The power performance of the joint U is compared with the two simpler statistics without either genomic or vertex kernel function. The performance under 8 sample size setting and the 4 scenarios of effect composition is shown in Figure [fig:PWR\_CNT\_KNL].

[fig:PWR\_CNT\_KNL] 

The top row of Figure [fig:PWR\_CNT\_KNL] shows that, the two parsimonious statistics and performed the best when the underlying effects were indeed purely genomic and vertex backed, respectively, but they are completely powerless when the actual effect compostion do not concure with their choice of U-kernel functions. In constrast, the joint statistic performed fairly well under both circumstances, close to the optimal power displayed by correlty specified parsimonious models. The bottom row of Figure [fig:PWR\_CNT\_KNL] shows joint U statistic outperformed both and when the effect is additve, either with (Figure [fig:PWR\_CNT\_KNL] down left) or without (Figure [fig:PWR\_CNT\_KNL] down right) an additional interaction term.

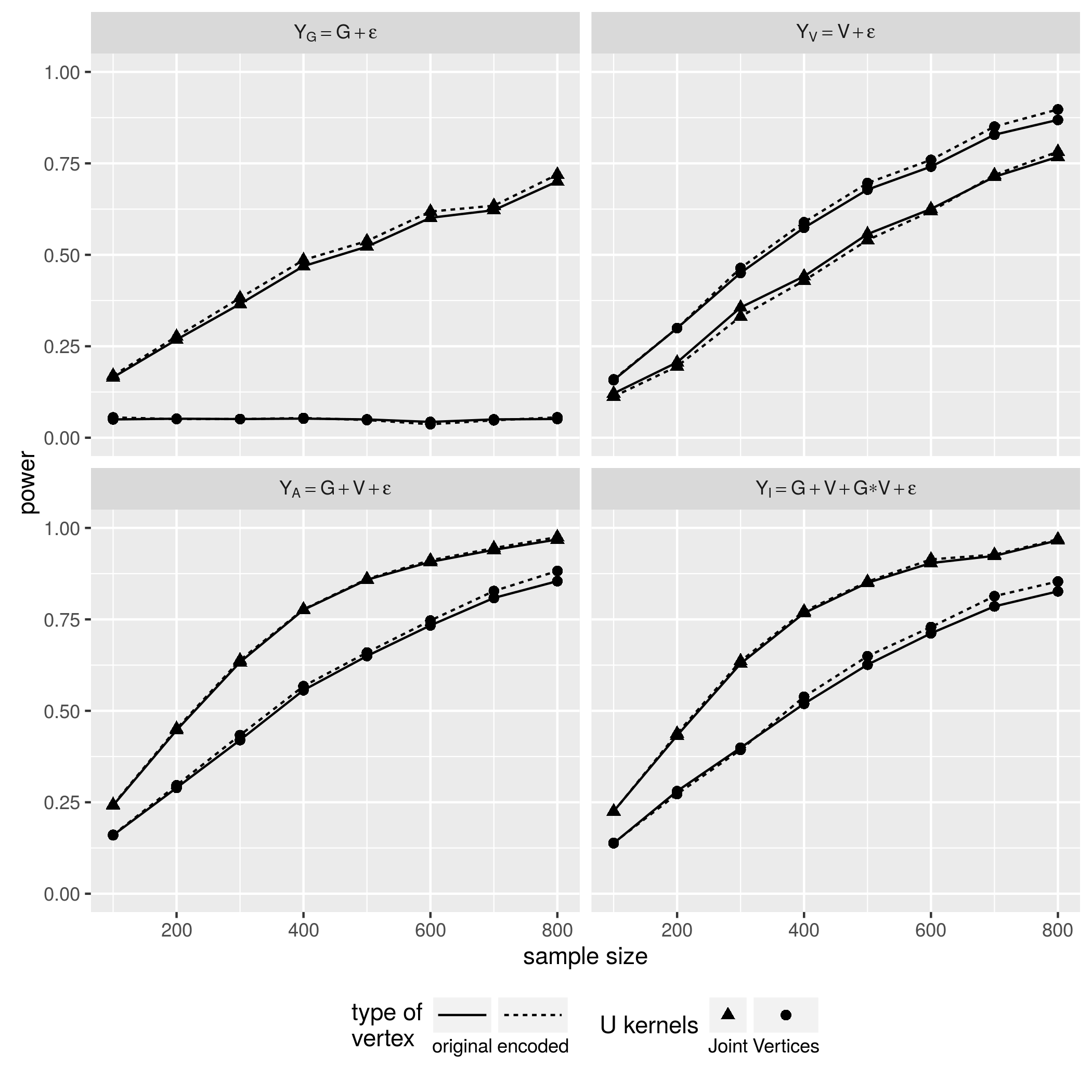
**Grouping and Aggregation on Vertices**  
We known vertices in a cortax profile do not have “low MAF” issue that rare genomic variants have, but the grouping and aggregation strategy used by analysis of NGS data may still benefit analysis involving cortial surfaces. The second study aims to see weather an aggregated cortical testing unit achieve higher power than the per-vertex based VWA followed by FDR (false discovery rate) correction. Comparison of the two strategy is done under the same 8 sample size and 4 effect compositions, but without the statistics since it does not involves cortex profile. The result is shown in Figure [fig:PWR\_CNT\_VWA].

[fig:PWR\_CNT\_VWA] 

Under all simulation settings, the aggregated testing unit (solid lines) overpowers the per-vertex based VWA (dashed lines). This gap is only slightly closed when the sample size grows large. Another interesting speculation is when the U kernel functions is totally mis-specified, the type I error rate of VWA is lower then the 0.05 threshold while the aggregated test isn’t (Figure [fig:PWR\_CNT\_VWA], top left panel). The multiple testing correction is done by false discover rate (FDR) adjustment, which says that the adjusted p-value will be conservative if the tests were not independent. Therefore, a conservative type I error rate refects the fact that closely located vertices are correlated as they were sampled from tightly connected brain tissue. As a result, grouping and signal aggregation is also recommended for cortex profile.

Another issue worth mentioning is the performance issue. The VWA takes drastically more time than the grouping and aggregation approach, because it has to perfrom tests in order to derive one final or statistic instead of just once. For now, the simulation study fixes to 512, under the largest sample size, 800, a single run of the proposed method using 1 cpu core takes only a few seconds, but the VWA approach paralleled on 4 cpu cores requires nearly 4 hours.

**Abstract High Order Features from Cortex Vertices** The third set of study tests whether the high order features abstracted from the raw cortex profile provides higher statistical power than the raw profile itself. Again the comparisons is done under all settings except those involving that doesn’t rely on vertices in the cortex. The result is shown in Figure [fig:PWR\_CNT\_SAE].

[fig:PWR\_CNT\_SAE] 

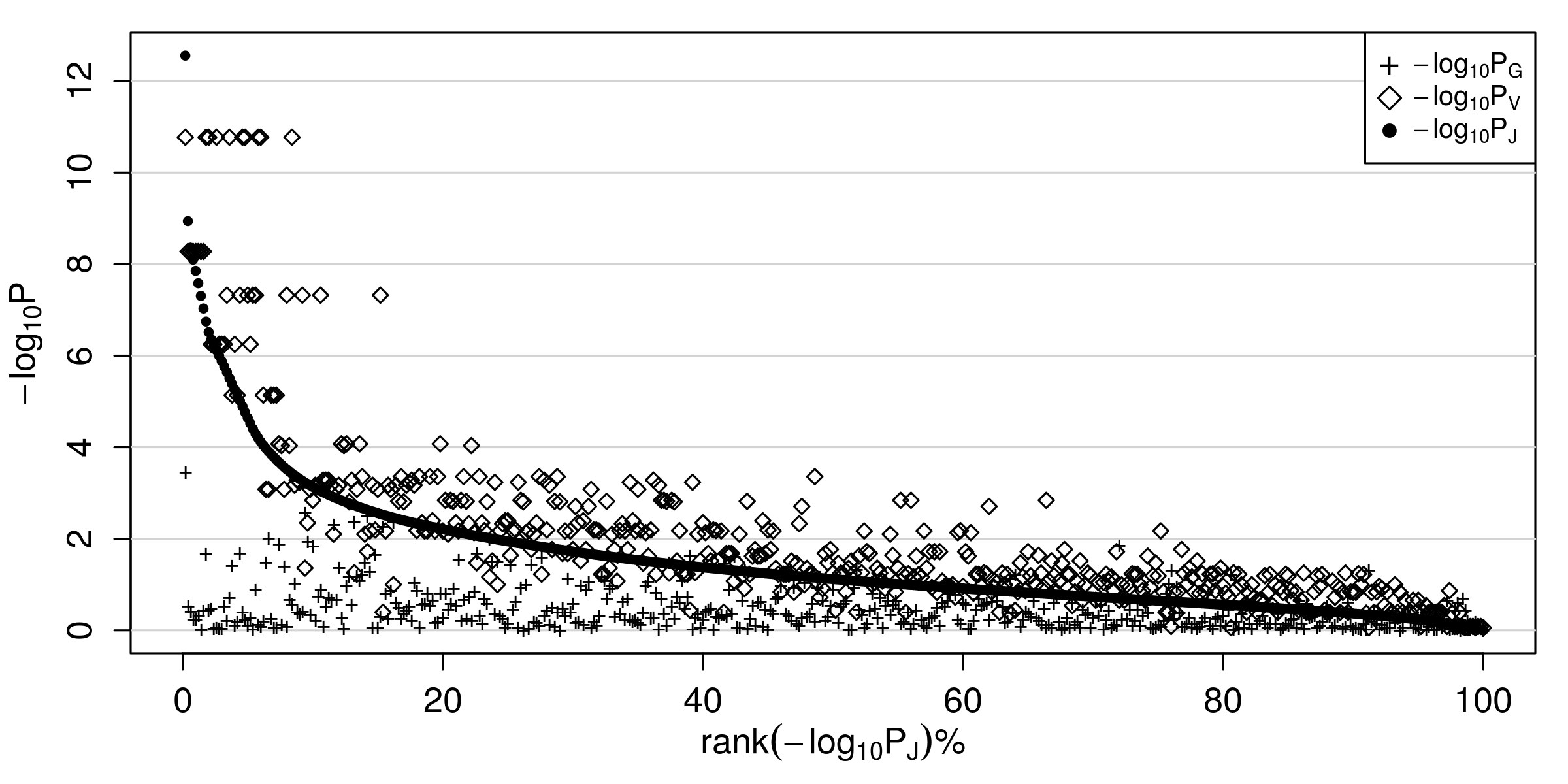
In most scenarios, the abstracted features (Figure [fig:PWR\_CNT\_SAE], dashed lines) offers more power then the orignal vertices (Figure [fig:PWR\_CNT\_SAE], solid lines), and the edge is growing with the sample size become larger. The only exception happened when the sample size is lower then 600, the effect is purely vertex based, and the partially mis-specified joint U statistic is used for the test (Figure [fig:PWR\_CNT\_SAE], top right). Since the rejection of null hypothesis counts as type I error when the kernel functions are complete mis-sepcified, the top left panel in [fig:PWR\_CNT\_SAE] shows that, the use of abstracted features doesn’t deviate the type I error rate from the 0.05 threshold.

**Simulation study of Binary phenotypes** The dichotomous responses is simulated by pluging the continuous responses into inverse logit function to get an array of probabilities, then draw the binary case/control status from these probabilities. The power performace shares very similar patterns under every scenario, albeit poorer then its continuous counterpart. The results is shown in Figure [fig:PWR\_BIN\_KNL], [fig:PWR\_BIN\_VWA] and [fig:PWR\_BIN\_SAE].

In general, the simulation studies have so far demenstrated the robustness and versitility of the proposed method when faced with uncertain effect composition and a variety of phenotype distributions. Also shown is the helpfulness of groupping and aggregation strategy used by many rare genomic variant studies over other types of high dimenstional whose variants are not “rare” but potentlly correlated. The power boost offered by the stacked autoencoders is not dramatic, but is increasingly more positive when the sample size grows.

# Real Data Analysis

The baseline data of 327 out of 806 participants who has definite diagnosis status entered the analysis. The genomic testing units are still defined by gene. The image testing units are now 68 cortical anatomy regions except bankssts in the right hemispheres. These 67 sets of vertices are sent to 67 cooresponding stacked autoencoders trained with all 806 profiles, the resulting 67 sets of high order features are then combined with the genes to perform the joint U statistic test. The total number of joint U statistics is , that is, the number of genes times the number of cortical regions. Among the 327 choosen subjects, 47 of them are diagnosed with either Alzheimer’s disease (AD) or dementia, while the rest 280 subjests are healthy controls (CN). The case/control outcomes were first regressed on 7 known risk factors of AD, namely age, gender, race, ethnicity, years of education, marriage status, ever smoking, and APOE 4 haplotype. The regression residuals were then taken as the actual phenotype. For each tuple of gene and cortical region, we also derive and to test the two simplified null hypothesis. Thus, the results came in triplets of p-values , and , corresponding to the three U statistics , , and . After negative log transformation, the triplets are show in Figure [fig:RDA\_PVL] horizontally, order by the transform .



p-values of real data analysis

In general, the significance of the purely vertex based (Figure [fig:RDA\_PVL] diamonds) is above the purely genomic based (Figure [fig:RDA\_PVL] crosses), reflecting the fact that genomic effect is weak while the cortex profile is a very indicator of the diseases in brain, and the joint similarity U statistic (Figure [fig:RDA\_PVL] dots) lies between the two, leaning closer to the cortical vertex based . To be noted is how “borrow” information from the cortex profile to enhance the statistical significance of the purely genomic based , which by itself never reaches significance at the threshold of after the FDR adjustment of tests. When is also moderately significant, the corresponding joint statistic could be more significant than both and , reaching the 0.05 threshold even after FDR adjustment over tests, which is shown by the dots in the most top left corner of Figure [fig:RDA\_PVL], and reflected by the 20 top significant triplets listed in Table [tab:RDS\_T20].

top 20 most significant joint test - overall

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GENE | CORTEX |  |  |  |  |  |
| IGLV1-44 | l.superiortemporal | 7271 | 174 |  |  |  |
| NBEAP2 | l.superiortemporal | 7271 | 238 |  |  |  |
| RPL21P89 | l.superiortemporal | 7271 | 90 |  |  |  |
| LOC102724504 | l.superiortemporal | 7271 | 59 |  |  |  |
| CNTNAP3P8 | l.superiortemporal | 7271 | 40 |  |  |  |
| CDH4 | l.superiortemporal | 7271 | 9464 |  |  |  |
| HNRNPA1P19 | l.superiortemporal | 7271 | 17 |  |  |  |
| FAM72C | l.superiortemporal | 7271 | 174 |  |  |  |
| RP11-638L3.1 | l.superiortemporal | 7271 | 4067 |  |  |  |
| CPXM1 | l.superiortemporal | 7271 | 208 |  |  |  |
| LOC101929612 | l.superiortemporal | 7271 | 256 |  |  |  |
| LOC100996517 | l.superiortemporal | 7271 | 34 |  |  |  |
| IGLV5-45 | l.superiortemporal | 7271 | 179 |  |  |  |
| MIS18BP1 | l.superiortemporal | 7271 | 553 |  |  |  |
| CDR2 | l.superiortemporal | 7271 | 260 |  |  |  |
| RPL41P2 | l.superiortemporal | 7271 | 87 |  |  |  |
| LOC101927737 | l.superiortemporal | 7271 | 157 |  |  |  |
| IGLV1-47 | l.superiortemporal | 7271 | 138 |  |  |  |
| IGLV7-46 | l.superiortemporal | 7271 | 130 |  |  |  |
| ZDHHC15 | l.superiortemporal | 7271 | 80 |  |  |  |

From Table [tab:RDS\_T20] we see the top 20 most significant test all involves the left superiortemporal cortex, whose neuron loss and shrinkage in volume is highly associated with the onset of Alzheimer’s Disease and its progression to dementia Gómez-Isla et al. (1997).

To see some more diverse cases of cortex profile “lend” information to the genomic test to enchance the significance of joint statistic , for each of the 67 cortical region find out the most significant involving that region, and listed the top 20 in Table [tab:RDS\_JNT].

top 20 most significant joint test - per cortical region

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CORTEX | GENE |  |  |  |  |  |
| IGLV1-44 | l.superiortemporal | 7271 | 174 |  |  |  |
| ZNF749 | l.entorhinal | 1102 | 321 |  |  |  |
| FAM72C | r.superiortemporal | 6868 | 174 |  |  |  |
| ZNF749 | r.entorhinal | 902 | 321 |  |  |  |
| FAM72C | l.cuneus | 1630 | 174 |  |  |  |
| ZNF749 | l.fusiform | 4714 | 321 |  |  |  |
| FAM72C | l.middletemporal | 4452 | 174 |  |  |  |
| FAM72C | r.cuneus | 1638 | 174 |  |  |  |
| ZNF749 | l.temporalpole | 839 | 321 |  |  |  |
| FAM72C | r.precuneus | 7975 | 174 |  |  |  |
| HSPD1P13 | l.pericalcarine | 1912 | 86 |  |  |  |
| FAM72C | r.fusiform | 4661 | 174 |  |  |  |
| HSPD1P13 | r.pericalcarine | 1823 | 86 |  |  |  |
| FAM72C | r.precentral | 10705 | 174 |  |  |  |
| HSPD1P13 | r.paracentral | 3831 | 86 |  |  |  |
| ZNF749 | r.temporalpole | 817 | 321 |  |  |  |
| FAM72C | l.precentral | 10740 | 174 |  |  |  |
| FAM72C | l.superiorfrontal | 12179 | 174 |  |  |  |
| FAM72C | l.postcentral | 9519 | 174 |  |  |  |
| ZNF749 | l.insula | 5229 | 321 |  |  |  |

We see in some cases both the genome and cortex based and do not reach statistical significance after multiple testing adjustment yet the joint statistic does, even if the number of tests () is way larger than the number of genes () or cortical regions (). These result suggest the existance of strong interaction of unknown type between the corresponding gene and cortex. Because the test is backed by grouping and signal aggregation, also the abstracted features which is 16 times smaller than the orignal cortex are used, the entire analysis of triplets can be quickly done in 12 hours in the MSU HPCC clusters.

# discussion

The proposed method can effectively combine information from multiple high dimensional data sources of distinct type to achieve higher statistical power, which in our case are the joint signal of genomic and cortex profiles. The major strength of the metod is its robustness and versatility. The robustness is demenstrated by its ability of retaining power that is close to the optimal model specification, when the true effect constitution was not known . The versatility is shown by its acceptance of a wide variety of profiles regardless of their distribution. The proposed method also applies variant grouping and signal aggregation strategy to the cortex profiles, which not only considerably boosted the statistcial power over the per-varaint screening procedure, but also save the computation time. With properly assigned kernel functions, the method can also incorporate additional profiles into the analysis, such as other “omics” data closer to the upstream genomics, or inflammatory bio-markers closer to the down-stream health outocme, without worrying about many possible high order interactions among more than two or three types of profile.

The method also build one of its components with the high order features abstracted from the cortex profile of eligible samples () using the stacked autoencoders (SA) tained with the whole dataset (). The abstracted features not only has lower dimensionality, but also help the methods to achieve slightly higher statistical power. We feel the potent of the deep artifical neural network is only explored at its surface, and the usefulness of the SA opened up intriguing prospects. First, the encoders can go deeper, by increasing the number of layers. Though a deeper SA is harder to train, it is capable of creating more compact yet more meaningful abstraction from the orignal input, subsequently boost the statistical power even further yet lower the computation load. Second, as mentioned before, the SA was trained in an unsupervised manner, which means a bank of data collected not for any particular study can be used to increamentally refine the existing SA, as long as the data collection follows compatible protocols. In our case, the 489 subjects who couldn’t enter the real data analysis due to uncertainly in diagnosis, still contributed their cortex profile to train the SAs. Further, in the near future, the rest of the ADNI participants who were not included in this study due to the lack of next generation sequencing genomic profile, can still help to refine the SAs we already have, because every ADNI participants has structure MRI data. Lastly, it is very tempting to use the same training and abstraction approach on the genomic profile, which is also high dimentional and growing in size, and again, with unsupervised training technique, we can utilize a huge wealth of NGS data from collaborators and public database, such as the freely available 1000 genome project data.

These prospects, are not coming without challenges. First regarding the construction of the joint U statistic, it is easy to bring in more high dimensional components, but when an overall association is detected, it is very hard to tease out exactly how each component contributed to that association, and it is also hard to tell the (other than a simple product form in the simulation) and effect size of that interaction between the components when more than two kernel functions are involved. In the real data analysis, when the joint U statistic turned out to be statistically significant while the two simplified and didn’t, the interaction between the genomic and cortex profile is guaranteed, but one can not be sure this interaction is associated with the phenotype profile. When all three statistics are significant, we known from the simulation study the interaction between genomic and cortex profiles may, or may not exists. The proposed method is better suited for fast screening of a large number of combinations of multiple high dimensional profiles, but in the end an explicit modeling is still reqiured for the categorization and quantifying of the associations.

Another challenge involves the way variants are grouped, and the way stacked autoencoders are trained and utilized. An autoencoder require constant input dimension. The cortex profile is stable in the number of vertices (), so are the 67 anatomy regions, if the current and furture samples are registered to the same atlas. We trained 67 stacked autoencoders for each region, but in reality, they are far too coarse to accurately pinpoint the loci in the cortex. For the genomic data, grouping variants by gene is an accepted compromise between accuracy and statistical power, however, the input dimension is not fixedsince the number and location of variants can differ from study to study, also the dimension of a gene can differ from sample to sample due to the indels and copy number variation. Besides, training over thousands stacked autoencoders is hardly affordable given the intensity of computation even if the greedy layer-wise pretrain techneque is used. So a grouping and training scheme not colluded with any existing functional information is required. Instead of training an SA for any specific gene or cortical region, an SA will be only be trained for an resonable input dimensionality, which is small enough to both satisfy the desired accuracy and ease the computation, but not too small so no meaningful high order feature is contained. Taking the cortex profile as an example, if the manipulation of vertices in the 3D space is not too complicated, one could first realign the vertices to a sphere with uniform spacing without altering the topology of the cortex, after which only one SA will be trained to encrypt the general knowledge of any cortical region mapped to a sphere surface of, for example, 5 degree in both latitude and longitude. For the genomic profile, only one SA will be trained to encrypt the general knowledge of, for example, a 100kb segment any where in the genome except the proximity of centromeres and telomeres. Yet, another complication will surface due to the inclusion of redundant sequences between the nucleotide polymorphisms, making the abstracted feature likely to have even higher dimension than the polymorphism in any 100kb region, unless a very deep stacked autoencoder is successfully trained. What can be guaranteed however, is the abundance of training materials.

There is also rooms to improve the simulation studies. We known the disease in CNS does not alter the genome, but it does so to the cortex, often making visible changes of features. The current simulation only assigned effects to vertices randomly chosen across a testing unit, instead of changing the thickness for a group of connected vertices to creat a visible shrinkage. As an result, the benefit of feature abstraction may be more pronounced in the real data analysis, without the evidence from the simulation study. The desired simulation however, may required intense human input because the change of features must mimic the real life clinical experiences, plus the difficulty to manipulate the vertices in 3D space. A more approachable simulation can be done by assigning the effect to vertices clusterd in existing visible featuers such as the few dozen named sulci and gyri which is already marked by [**FreeSurfer**](http://surfer.nmr.mgh.harvard.edu).

Should these challenges be addressed and improvements be realized, we see a greater used of the rich and ever accumulating data, for more powerful inference of the relationship between complex disease and genome.

# Appendix

## Gradient Descent

The optimization is done by gradient decent. Starting with a randomly initialized assignment of at , and compute the next assignment by substracting from the current assignment a small fraction of the gradient of reconstruction loss with respect to the current assignment (). The small fraction is called a learning step, if the step is reasonablly small, the reconstruction loss will keep dropping. The learning will repeat until cease to drop. The final assignment is considered the optimal .

The gradient is calculated by a process called backward propagation [?], which relies heavily on the chain rule of derivatives. First, realize the total loss is a summation of of individual loss for , and in turn is a function of (the observed is a constant vector), at last, is a function of all structure parameters , using the additive rule and chain rule we get:

Here we use to denote element-wise division. From now on we will focus on the th. sample and ommit from the subscript, since the output on top is constructed in exactly the same manner across all individuals. Recall the symbolic form of SA and the decoders on its top (see [eq:ES], [eq:DS] and [eq:ED]), the output of any layer, either from an encoder or a decoder, is a function of its own structure parameters and the direct input come from the lower layer. The gradient of the layer output with respect to its own paremeters and input can be calculated directly. Let us take the th. decoder layer as an representative, whose own ouput, parameters, and input are , and , respectively (see [eq:DS]). The calculation for an encoder layer will be similar, since the structure of an encoder layer is identical to its decoder counterpart except the dimensionality change (see [eq:ES] and [eq:ED]). The derivatives will be calculated separately for each output element , because the inverse logit transformation is applied element-wise. To ease the thought process, the th. decoder layer is rewritten in the per-element manner,

where is the th. row vector of weight matrix , and is the th. element of threshold vector , with . The gradients of the th. element in the output, , with respect to its contributing parameters and the input are

respectively. Notice that, the derivative of has a more compact expression using the dependent variable instead of , since

thus, during the invocation of chain rule, we have

Packing up the gradient for each element in the th. decoder’s output into a vector, we have

$$\begin{split}
\arraycolsep=1.4pt\def\arraystretch{1.5}
{\frac{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}}}{\partial {\boldsymbol{\tilde{\theta}}}\_{i}}}
&= \left[\begin{array}{c}
{\frac{\partial \tilde{z}\_{i-1,1}}{\partial [{{{{\boldsymbol{\tilde{w}}}}\_{i1}^{1 \times d\_{i}}}}, \tilde{b}\_{i1}]}} \\
{\frac{\partial \tilde{z}\_{i-1,2}}{\partial [{{{{\boldsymbol{\tilde{w}}}}\_{i2}^{1 \times d\_{i}}}}, \tilde{b}\_{i2}]}} \\ \vdots \\
{\frac{\partial \tilde{z}\_{i-1,k}}{\partial [{{{{\boldsymbol{\tilde{w}}}}\_{ik}^{1 \times d\_{i}}}}, \tilde{b}\_{ik}]}} \\ \vdots \\
{\frac{\partial \tilde{z}\_{i-1,d\_{i-1}}}{\partial [{{{{\boldsymbol{\tilde{w}}}}\_{id\_{i-1}}^{1 \times d\_{i}}}}, \tilde{b}\_{id\_{i-1}}]}} \\
\end{array} \right]
= \left[\begin{array}{c}
\tilde{z}\_{i-1,1}(1-\tilde{z}\_{i-1,1}) [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}}, 1] \\
\tilde{z}\_{i-1,2}(1-\tilde{z}\_{i-1,2}) [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}}, 1] \\ \vdots \\
\tilde{z}\_{i-1,k}(1-\tilde{z}\_{i-1,k}) [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}}, 1] \\ \vdots \\
\tilde{z}\_{i-1,d\_{i-1}}(1-\tilde{z}\_{i-1,d\_{i-1}}) [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}}, 1] \\
\end{array} \right] \\
&= {\text{diag}({{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {\text{diag}({\boldsymbol{1}}- {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}},1], \\ \text{in a similarly way,} \\
{\frac{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}}
&= \left[\begin{array}{c}
{\frac{\partial \tilde{z}\_{i-1,1}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}} \\
{\frac{\partial \tilde{z}\_{i-1,2}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}} \\ \vdots \\
{\frac{\partial \tilde{z}\_{i-1,k}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}} \\ \vdots \\
{\frac{\partial \tilde{z}\_{i-1,d\_{i-1}}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}} \\
\end{array} \right]
= \left[\begin{array}{c}
\tilde{z}\_{i-1,1}(1-\tilde{z}\_{i-1,1}){{{{\boldsymbol{\tilde{w}}}}\_{i1}^{1 \times d\_{i}}}} \\
\tilde{z}\_{i-1,2}(1-\tilde{z}\_{i-1,2}){{{{\boldsymbol{\tilde{w}}}}\_{i2}^{1 \times d\_{i}}}} \\ \vdots \\
\tilde{z}\_{i-1,k}(1-\tilde{z}\_{i-1,k}){{{{\boldsymbol{\tilde{w}}}}\_{ik}^{1 \times d\_{i}}}} \\ \vdots \\
\tilde{z}\_{i-1,d\_{i-1}}(1-\tilde{z}\_{i-1,d\_{i-1}}){{{{\boldsymbol{\tilde{w}}}}\_{id\_{i-1}}^{1 \times d\_{i}}}} \\
\end{array} \right] \\
&= {\text{diag}({{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {\text{diag}({\boldsymbol{1}}- {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {{{\boldsymbol{\tilde{W}}}}\_{i}^{d\_{i-1} \times d\_{i}}}
\end{split}$$

$$\label{eq:GD}
\begin{split}\
\begin{array}{rl}
\textrm{the i th. decoder:} & \begin{array}{rcl}
{\frac{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}}}{\partial {\boldsymbol{\tilde{\theta}}}\_i}} & = & {\text{diag}({{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {\text{diag}({\boldsymbol{1}}{} - {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} [{{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}\prime}}}, 1] \\
{\frac{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}}}{\partial {{{{\boldsymbol{\tilde{z}}}}\_{i}^{d\_{i}}}}}} & = & {\text{diag}({{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {\text{diag}({\boldsymbol{1}}{} - {{{{\boldsymbol{\tilde{z}}}}\_{i-1}^{d\_{i-1}}}})} {{{\boldsymbol{\tilde{W}}}}\_{i}^{d\_{i-1} \times d\_{i}}} \\
\end{array} \\ \\
\textrm{the i th. encoder:} & \begin{array}{rcl}
{\frac{\partial {{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}}}{\partial {\boldsymbol{\theta}}\_i}} & = & {\text{diag}({{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}})}{\text{diag}({\boldsymbol{1}}- {{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}})} [{{{{\boldsymbol{z}}}\_{i-1}^{d\_{i-1}\prime}}}, 1] \\
{\frac{\partial {{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}}}{\partial {{{{\boldsymbol{z}}}\_{i-1}^{d\_{i-1}}}}}} & = & {\text{diag}({{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}})} {\text{diag}({\boldsymbol{1}}- {{{{\boldsymbol{z}}}\_{i}^{d\_{i}}}})} {{{\boldsymbol{W}}}\_{i}^{d\_{i} \times d\_{i-1}}}\\
\end{array}
\end{array}
\end{split}$$

Here means creating a matrix and asign the vector to its diagonal. Since the input of a layer is essencially the output of the layer down below, who also has its own structure paremeters and input from the layer even lower, the gradient of the top output, , with respect to any lower layers’ parameters, can be calculated by recursively invoking the chain rule, that is,

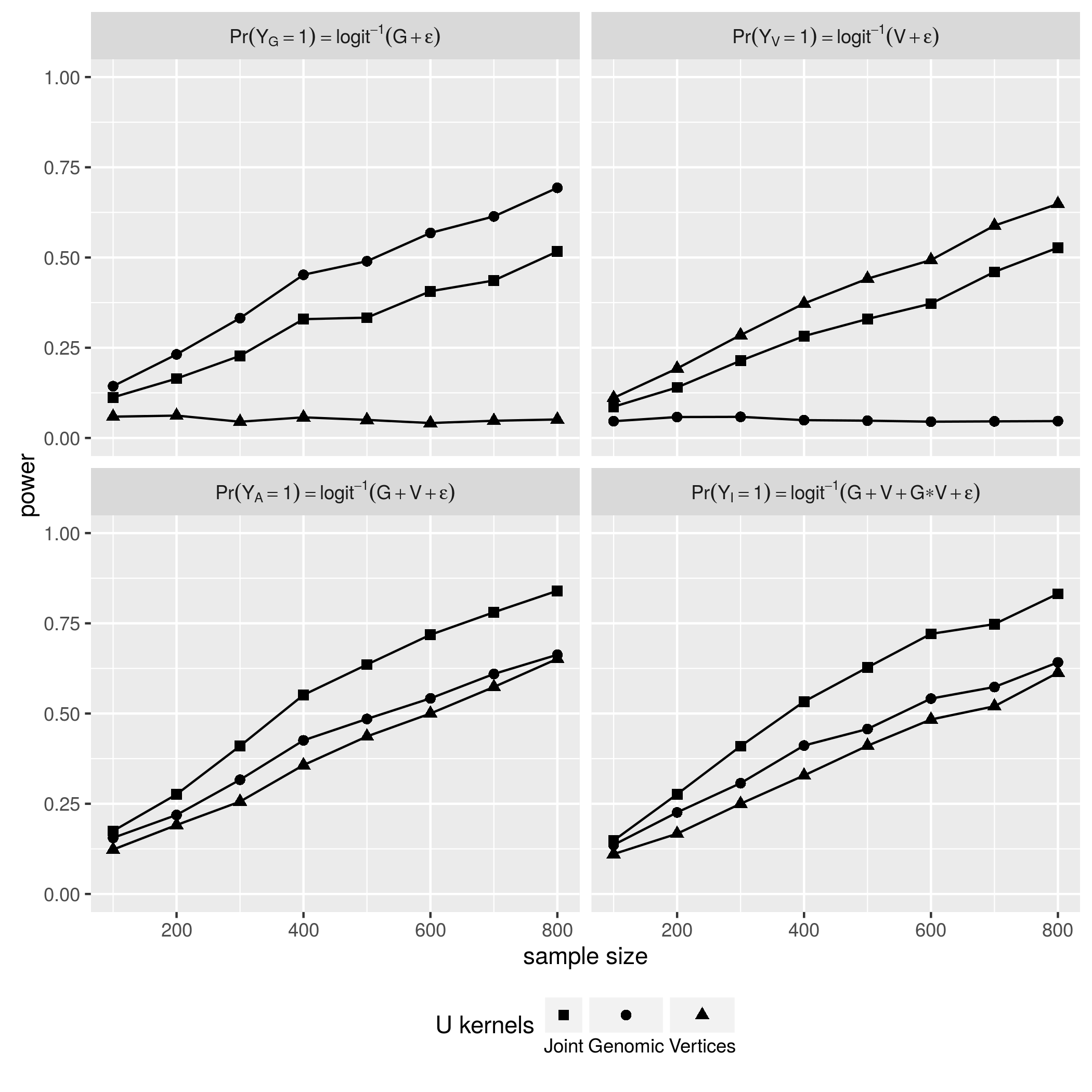
$$\label{eq:BP}
\newcommand{\PDT}[3]{{\frac{\partial #1}{\partial #2}}{\frac{\partial #2}{\partial #3}}}
\newcommand{\CHN}[3]{{\frac{\partial #1}{\partial #3}} & = & \PDT{#1}{#2}{#3}}
\newcommand{\SEP}{\quad \quad}
\arraycolsep=1.4pt\def\arraystretch{1.4}
\begin{array}{rclcrcl}
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{1 }}{{\boldsymbol{\tilde{\theta}}}\_{2 }} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{1 }}{{{\boldsymbol{\tilde{z}}}}\_{2 }} \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{2 }}{{\boldsymbol{\tilde{\theta}}}\_{3 }} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{2 }}{{{\boldsymbol{\tilde{z}}}}\_{3 }} \\
& \vdots & & \SEP & & \vdots & \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{M-2}}{{\boldsymbol{\tilde{\theta}}}\_{M-1}} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{M-2}}{{{\boldsymbol{\tilde{z}}}}\_{M-1}} \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{M-1}}{{\boldsymbol{\tilde{\theta}}}\_{M }} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{\tilde{z}}}}\_{M-1}}{{{\boldsymbol{\tilde{z}}}}\_{M }} \\
& & & & {\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {{\boldsymbol{z}}}\_{M }}} & = & {\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {\boldsymbol{\hat{y}}}}} = {\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {{\boldsymbol{\tilde{z}}}}\_M}} \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{M }}{{\boldsymbol{\theta}}\_{M }} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{M }}{{{\boldsymbol{z}}}\_{M-1}} \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{M-1}}{{\boldsymbol{\theta}}\_{M-1}} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{M-1}}{{{\boldsymbol{z}}}\_{M-2}} \\
& \vdots & & \SEP & & \vdots & \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{2 }}{{\boldsymbol{\theta}}\_{2 }} & \SEP & \CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{2 }}{{{\boldsymbol{z}}}\_{1 }} \\
\CHN{{{\boldsymbol{\tilde{z}}}}\_0}{{{\boldsymbol{z}}}\_{1 }}{{\boldsymbol{\theta}}\_{1 }} & \SEP & & &
\end{array}$$

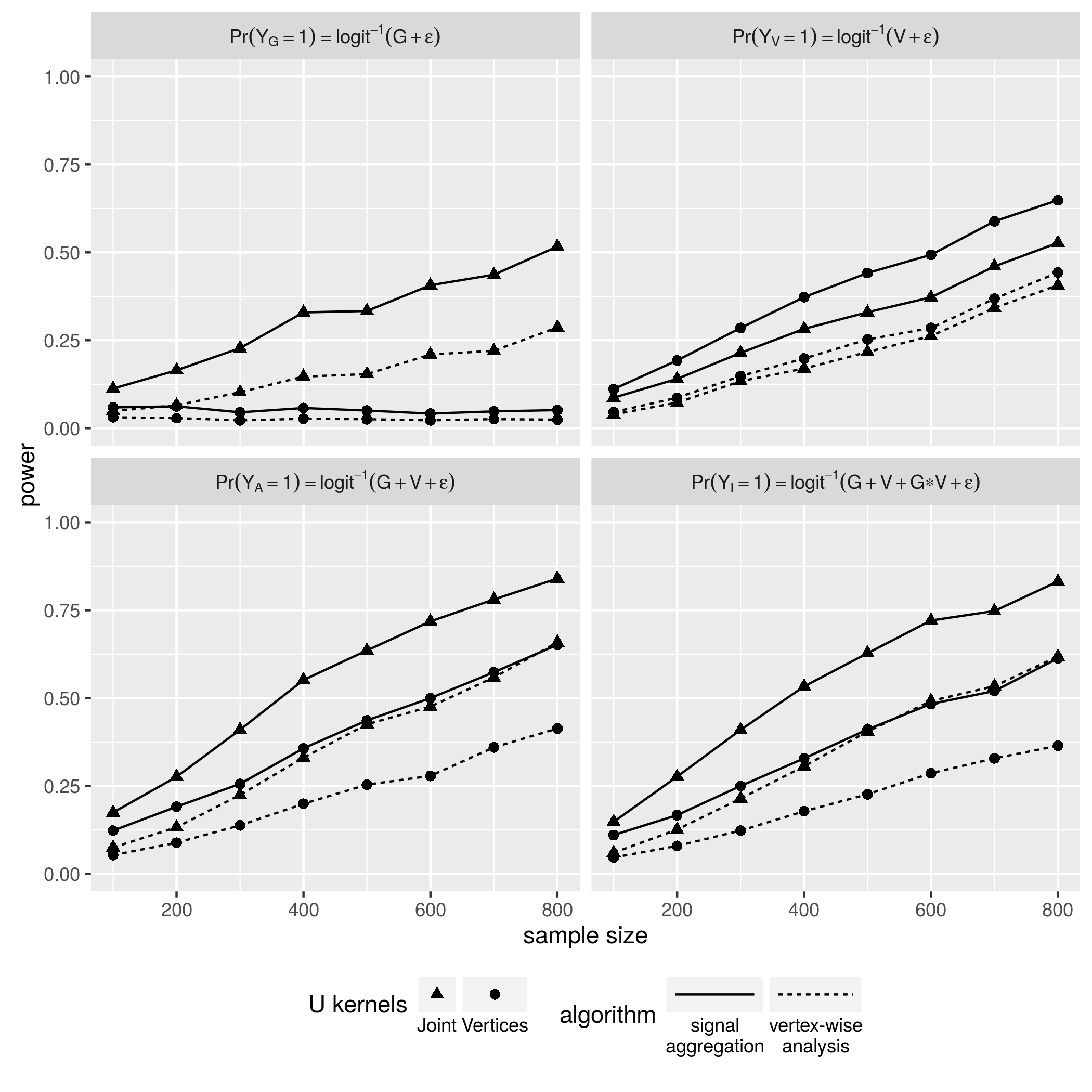
In reverse to the “bottom to top” encoding and decoding procedure, the gradient is propagated from top to bottom, hince the name “backward propagation”. Taking the layered SA and its decoder counterpart together, the total number of parameters to be calibrated is . The optimization can be computationally intense because this number is usually huge. One commonly applied strategy for learning an SA is to constrain the weight matrix in the decoder layers to be the transpose of their counterpart in the SA, that is, forcing for , and on top of the assignment of gradient layed out in [eq:GD] and [eq:BP], the gradient of the top output with respect to the weight matrix of any decoder will be absorded by the one in its encoder counterpart, that is,

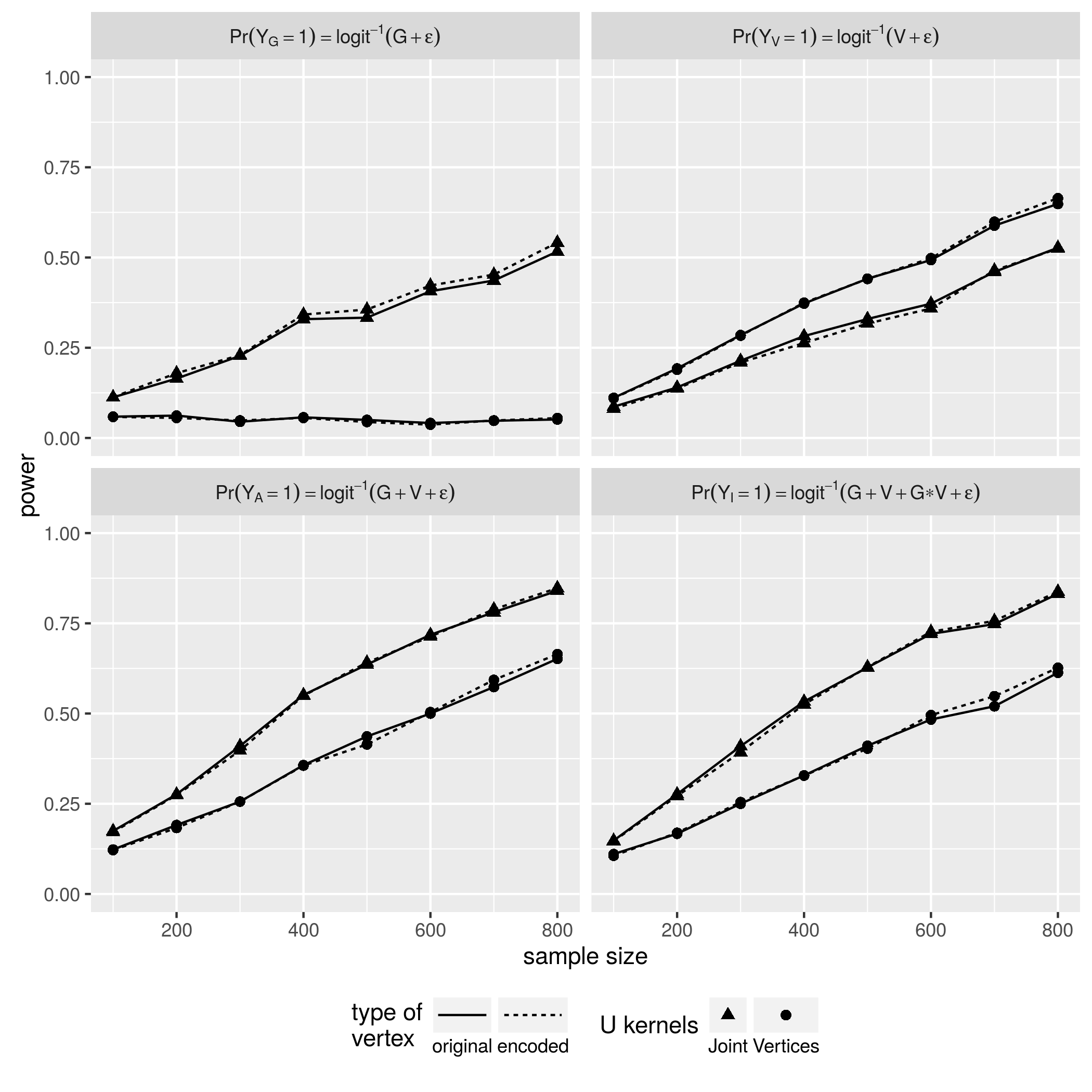
$$\begin{split}\
{\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {{\boldsymbol{W}}}\_i}}^\* &= {\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {{\boldsymbol{W}}}\_i}} + ({\frac{\partial {{\boldsymbol{\tilde{z}}}}\_0}{\partial {{\boldsymbol{W}}}\_i}})^\prime.
\end{split}$$

Doing so introduces slightly more computation for each learning step, but at the same time almost halve the number of tuning parameters to , greatly speed up the convergence of . Beside, the whole structure fits the common sense that, encoding and decoding are essentially symmetric operations. More importantly, the constraint encourages learning of an optimal SA instead of a sub-optimal SA coupled with a powerful decoder on its top, afterall, our best interest is the high order feature abstracted from the raw input, not its reconstruction.

## Simulation study of Binary Phenotype

[fig:PWR\_BIN\_KNL] 

[fig:PWR\_BIN\_VWA] 

[fig:PWR\_BIN\_SAE] 

Ashburner, John, and Karl J. Friston. 2000. “Voxel-Based Morphometry—The Methods.” *NeuroImage* 11 (6): 805–21. doi:[http://dx.doi.org/10.1006/nimg.2000.0582](https://doi.org/http://dx.doi.org/10.1006/nimg.2000.0582).

Baron, J.C., G. Chételat, B. Desgranges, G. Perchey, B. Landeau, V. de la Sayette, and F. Eustache. 2001. “In Vivo Mapping of Gray Matter Loss with Voxel-Based Morphometry in Mild Alzheimer’s Disease.” *NeuroImage* 14 (2): 298–309. doi:[http://dx.doi.org/10.1006/nimg.2001.0848](https://doi.org/http://dx.doi.org/10.1006/nimg.2001.0848).

Bengio, Yoshua, Pascal Lamblin, Dan Popovici, Hugo Larochelle, and others. 2007. “Greedy Layer-Wise Training of Deep Networks.” *Advances in Neural Information Processing Systems* 19. MIT; 1998: 153.

Bernal-Rusiel, Jorge L., Douglas N. Greve, Martin Reuter, Bruce Fischl, and Mert R. Sabuncu. 2013. “Statistical Analysis of Longitudinal Neuroimage Data with Linear Mixed Effects Models.” *NeuroImage* 66: 249–60. doi:[http://dx.doi.org/10.1016/j.neuroimage.2012.10.065](https://doi.org/http://dx.doi.org/10.1016/j.neuroimage.2012.10.065).

Chételat, G., B. Landeau, F. Eustache, F. Mézenge, F. Viader, V. de la Sayette, B. Desgranges, and J.-C. Baron. 2005. “Using Voxel-Based Morphometry to Map the Structural Changes Associated with Rapid Conversion in Mci: A Longitudinal {Mri} Study.” *NeuroImage* 27 (4): 934–46. doi:[http://dx.doi.org/10.1016/j.neuroimage.2005.05.015](https://doi.org/http://dx.doi.org/10.1016/j.neuroimage.2005.05.015).

Cirulli, Elizabeth T, and David B Goldstein. 2010. “Uncovering the Roles of Rare Variants in Common Disease Through Whole-Genome Sequencing.” *Nature Reviews Genetics* 11 (6). Nature Publishing Group: 415–25.

Dai, Hongying (Daisy), J Steven Leeder, and Yuehua Cui. 2014. “A Modified Generalized Fisher Method for Combining Probabilities from Dependent Tests.” *Frontiers in Genetics* 5 (32). doi:[10.3389/fgene.2014.00032](https://doi.org/10.3389/fgene.2014.00032).

Fischl, Bruce. 2012. “FreeSurfer.” *NeuroImage* 62 (2): 774–81. doi:[http://dx.doi.org/10.1016/j.neuroimage.2012.01.021](https://doi.org/http://dx.doi.org/10.1016/j.neuroimage.2012.01.021).

Glorot, Xavier, and Yoshua Bengio. 2010. “Understanding the Difficulty of Training Deep Feedforward Neural Networks.” In *International Conference on Artificial Intelligence and Statistics*, 249–56.

Gómez-Isla, Teresa, Richard Hollister, Howard West, Stina Mui, John H. Growdon, Ronald C. Petersen, Joseph E. Parisi, and Bradley T. Hyman. 1997. “Neuronal Loss Correlates with but Exceeds Neurofibrillary Tangles in Alzheimer’s Disease.” *Annals of Neurology* 41 (1). Wiley Subscription Services, Inc., A Wiley Company: 17–24. doi:[10.1002/ana.410410106](https://doi.org/10.1002/ana.410410106).

Hinton, G. E., and R. R. Salakhutdinov. 2006. “Reducing the Dimensionality of Data with Neural Networks.” *Science* 313 (5786). American Association for the Advancement of Science: 504–7. doi:[10.1126/science.1127647](https://doi.org/10.1126/science.1127647).

Lawlor, Debbie A., Roger M. Harbord, Jonathan A. C. Sterne, Nic Timpson, and George Davey Smith. 2008. “Mendelian Randomization: Using Genes as Instruments for Making Causal Inferences in Epidemiology.” *Statistics in Medicine* 27 (8). John Wiley & Sons, Ltd.: 1133–63. doi:[10.1002/sim.3034](https://doi.org/10.1002/sim.3034).

Madsen, Sharon R., Bo Eskerod AND Browning. 2009. “A Groupwise Association Test for Rare Mutations Using a Weighted Sum Statistic.” *PLoS Genet* 5 (2). Public Library of Science: e1000384. doi:[10.1371/journal.pgen.1000384](https://doi.org/10.1371/journal.pgen.1000384).

Manolio, Teri A. 2010. “Genomewide Association Studies and Assessment of the Risk of Disease.” *New England Journal of Medicine* 363 (2): 166–76. doi:[10.1056/NEJMra0905980](https://doi.org/10.1056/NEJMra0905980).

Pandey, Janardan P. 2010. “Comment on Genomewide Association Studies and Assessment of Risk of Disease.” *New England Journal of Medicine* 363 (21): 2076–7. doi:[10.1056/NEJMc1010310](https://doi.org/10.1056/NEJMc1010310).

Purcell, Shaun, Benjamin Neale, Kathe Todd-Brown, Lori Thomas, Manuel AR Ferreira, David Bender, Julian Maller, et al. 2007. “PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses.” *The American Journal of Human Genetics* 81 (3). Elsevier: 559–75.

Salat, David H., Randy L. Buckner, Abraham Z. Snyder, Douglas N. Greve, Rahul S.R. Desikan, Evelina Busa, John C. Morris, Anders M. Dale, and Bruce Fischl. 2004. “Thinning of the Cerebral Cortex in Aging.” *Cerebral Cortex* 14 (7): 721–30. doi:[10.1093/cercor/bhh032](https://doi.org/10.1093/cercor/bhh032).

Smith, George Davey, and Shah Ebrahim. 2003. “‘Mendelian Randomization’: Can Genetic Epidemiology Contribute to Understanding Environmental Determinants of Disease?” *International Journal of Epidemiology* 32 (1). IEA: 1–22.

Smith, Stephen M, Mark Jenkinson, Heidi Johansen-Berg, Daniel Rueckert, Thomas E Nichols, Clare E Mackay, Kate E Watkins, et al. 2006. “Tract-Based Spatial Statistics: Voxelwise Analysis of Multi-Subject Diffusion Data.” *Neuroimage* 31 (4). Elsevier: 1487–1505.

Vincent, Pascal, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion.” *J. Mach. Learn. Res.* 11 (December). JMLR.org: 3371–3408. <http://dl.acm.org/citation.cfm?id=1756006.1953039>.

Wei, Changshuai, Robert C Elston, and Qing Lu. 2015. “A Weighted U Statistic for Association Analysis Considering Genetic Heterogeneity.” *ArXiv Preprint ArXiv:1504.08319*.

Wu, Michael C., Seunggeun Lee, Tianxi Cai, Yun Li, Michael Boehnke, and Xihong Lin. 2011. “Rare-Variant Association Testing for Sequencing Data with the Sequence Kernel Association Test.” *The American Journal of Human Genetics* 89 (1): 82–93. doi:[http://dx.doi.org/10.1016/j.ajhg.2011.05.029](https://doi.org/http://dx.doi.org/10.1016/j.ajhg.2011.05.029).