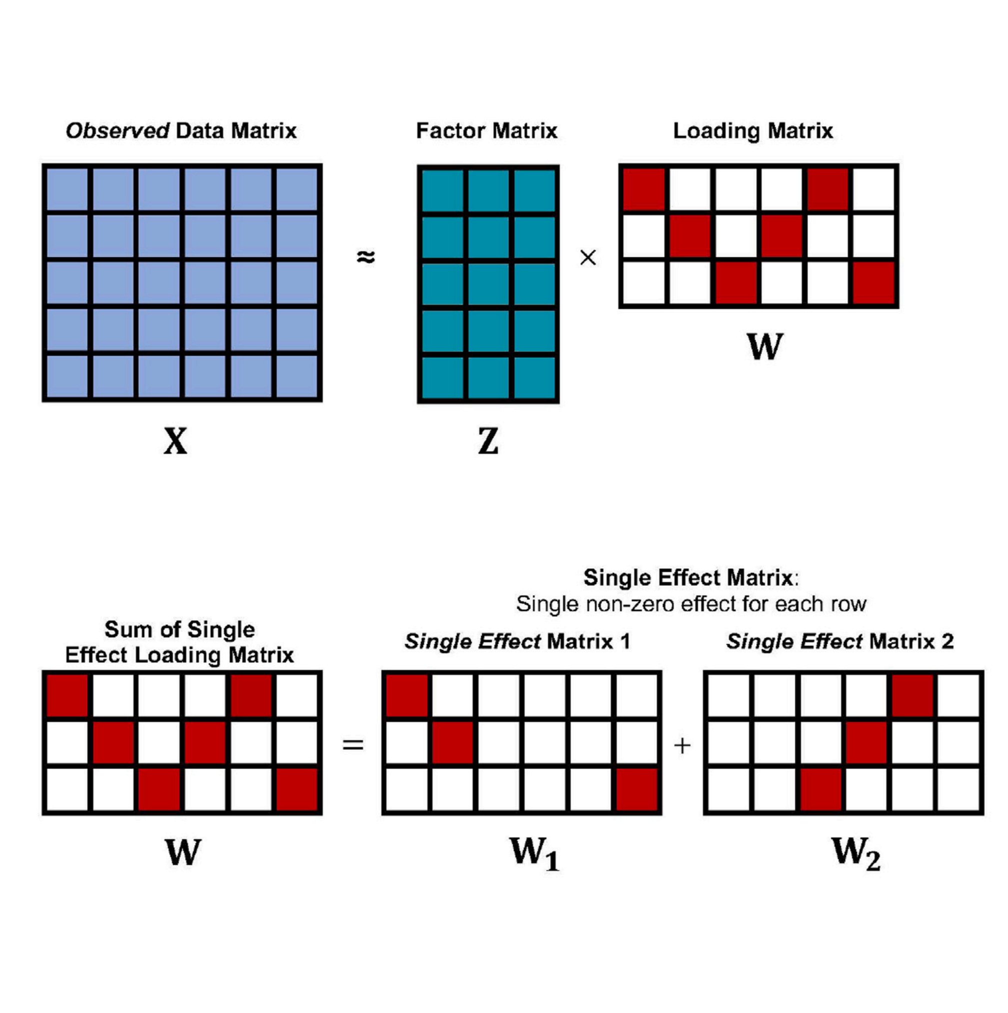
iScience

CellPress OPEN ACCESS

Article

SuSiE PCA: A scalable Bayesian variable selection technique for principal component analysis



Dong Yuan,
Nicholas Mancuso

dongyuan@usc.edu (D.Y.)
nmancuso@usc.edu (N.M.)

Highlights

Efficient PCA feature selection via posterior inclusion probabilities

Learned model prior enhances inferential robustness

Seamless CPU/GPU/TPU implementation enables efficient inference

Yuan & Mancuso, iScience *26*, 108181
November 17, 2023 © 2023 The Author(s).
https://doi.org/10.1016/j.isci.2023.108181

iScience

CellPress
OPEN ACCESS

Article

SuSiE PCA: A scalable Bayesian variable selection technique for principal component analysis

Dong Yuan[1,3,*] and Nicholas Mancuso[1,2,*]

SUMMARY

Latent factor models, like principal component analysis (PCA), provide a statistical framework to infer low-rank representation in various biological contexts. However, feature selection is challenging when this low-rank structure manifests from a sparse subspace. We introduce SuSiE PCA, a scalable sparse latent factor approach that evaluates uncertainty in contributing variables through posterior inclusion probabilities. We validate our model in extensive simulations and demonstrate that SuSiE PCA outperforms other approaches in signal detection and model robustness. We apply SuSiE PCA to multi-tissue expression quantitative trait loci (eQTLs) data from GTEx v8 and identify tissue-specific factors and their contributing eGenes. We further investigate its performance on the large-scale perturbation data and find that SuSiE PCA identifies modules with a higher enrichment of ribosome-related genes than sparse PCA (false discovery rate [FDR] = 9.2×10^{-82} vs. 1.4×10^{-33}), while being $\sim$ 18x faster. Overall, SuSiE PCA provides an efficient tool to identify relevant features in high-dimensional biological data.

INTRODUCTION

Principal component analysis (PCA) is a popular dimension reduction technique[1] that has been widely applied for exploratory data analysis in many fields. One notable functionality of PCA is to synthesize crucial information across features into a small number of principal components (PCs). For example, PCA is commonly used to infer population structure from large-scale genetic data.[2,3] The top PCs explain differences in genetic variation arising from different geographic origins and ancestry of individuals, due to historical migration, admixture, etc.[4] Moreover, PCA provides a means to rank contributing relevant variables for each latent component, as Tipping and Bishop (1986) proposed the probabilistic reformulation of principal component analysis (PPCA).[5] Specifically, each PC is independent of other PCs and has its unique weights to represent the "importance" of original features, suggesting different latent components arise from different combinations of variables, or distinct aspects of information from the data.

However, one disadvantage of conventional PCA is that PCs provide limited interpretability, as each results from a linear combination of variables in the data.[6] To improve the interpretability of PCs, while providing an identifiable solution in high-dimensional data, a common approach is to impose sparsity on the PCA loadings. Broadly speaking, there are two types of approaches to achieving sparsity on the loading matrix. The first is the regularization methods such as sparse PCA,[6] which rewrites the PCA as a regression-based optimization problem and then includes a L_1 penalty on the objective function to achieve sparse loadings. The second type of method is the Bayesian treatment of PPCA, which imposes sparsity-induced prior on the factor loading matrix.[7–12] Despite various methods that focus on inducing sparse solutions for PCA, few provide a statistically rigorous way to select variables relevant to each factor in a post hoc manner. Although several sparse models are capable of shrinking the loadings of uninformative variables to zero, for those variables with non-zero weights, neither a reasonable threshold nor a formal statistical test is provided to inform feature prioritization for validation or follow-up.

Here, we propose SuSiE PCA, a highly scalable Bayesian framework for sparse PCA, that quantifies the uncertainty of contributing features for each latent component. Specifically, SuSiE PCA leverages the recent "sum of single effects" (SuSiE) approach[13] to model a loading matrix such that each latent factor contains at most L contributing features. Latent factors and sparse loading weights are learned through an efficient variational algorithm. In addition to providing a sparse loading matrix, SuSiE PCA computes posterior inclusion probabilities (PIPs) for each feature, which enables defining $\rho -$ level credible sets for feature selection. We demonstrate through extensive simulations that SuSiE PCA outperforms sparse PCA[6] and empirical Bayes matrix factorization (EBMF)[12] in identifying relevant features contributing to structured data while being robust to data-generating assumptions. Next, we apply SuSiE PCA to multi-tissue expression quantitative trait loci (eQTLs) data from the the genotype-tissue expression (GTEx) v8[12,14] study to identify tissue-specific components of regulatory genetic features and contributing eGenes (genes that have an associated eQTL). We also apply SuSiE PCA to high-dimensional perturb-seq data (CRISPR-based

[1]Biostatistics Division, Department of Population and Public Health Sciences, Keck School of Medicine, University of Southern California, Los Angeles, CA 90089, USA
[2]Center for Genetic Epidemiology, Keck School of Medicine, University of Southern California, Los Angeles, CA 90089, USA
[3]Lead contact
*Correspondence: dongyuan@usc.edu (D.Y.), nmancuso@usc.edu (N.M.)
https://doi.org/10.1016/j.isci.2023.108181

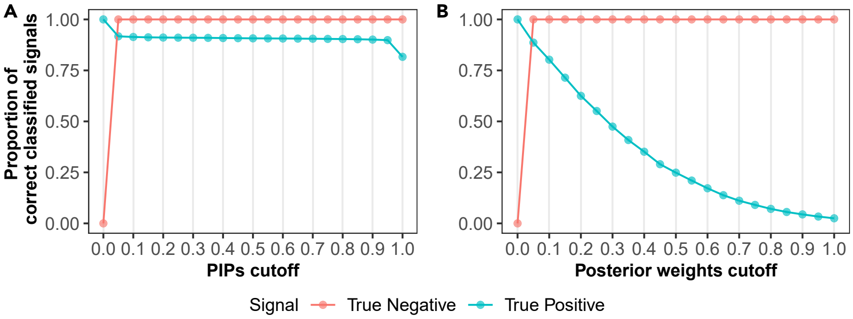


Figure 1. PIPs exhibit a higher efficiency in selecting the true signals than the posterior weights in SuSiE PCA

The proportion of correct classified signals using PIPs as cutoff (A) or posterior weights as cutoff (B). The green dots represent sensitivity, i.e., Pr(PIPs $\geq$ cutoff| True positive signal), and the red dots represent specificity, i.e., Pr(PIPs $<$cutoff| True false signal). For consistency and to ensure comparability between PIPs and weights, the weights are normalized to be ranged from 0 to 1.

screens with single-cell RNA-sequencing readouts)[15] and identify gene sets more enriched in the ribosome and coronavirus disease pathways when compared with sparse PCA (false discovery rate (FDR) = 9.2×10^{-82}, 63 genes involved vs. 1.4×10^{-33}, 35 genes involved) while requiring 17.8 times less computing time. Overall, we find that SuSiE PCA provides an efficient approach to compute interpretable latent factors from high-dimensional biological data. We provide an open-source python implementation that can run seamlessly on central processing unit (CPU), graphics processing unit (GPU), or tensor processing unit (TPU) available at http://www.github.com/mancusolab/susiepca.

RESULTS

PIPs from SuSiE PCA outperform existing approaches for PCA feature selection

To evaluate the performance of SuSiE PCA, we performed extensive simulations (see details in STAR Methods). Briefly, we performed 100 simulations by varying model parameters one at a time and performed inference using SuSiE PCA with the true number of latent variables (*K*) and effects (*L*) known. First, we evaluated the ability of inferred PIPs to discriminate between relevant and non-relevant features for latent factors. Specifically, we compared the sensitivity and specificity of inferred PIPs to normalized posterior mean weights from SuSiE PCA (see Figure 1). When selecting variables based on PIPs >0.90, SuSiE PCA identifies 88.9% of true positive (non-zero) signals, demonstrating largely calibrated posterior inference. We observed nearly all true negative signals exhibited PIPs <0.05. As a comparison, the normalized posterior weights performed well on excluding the true negative signals but failed to capture true positive signals as rapidly as PIP thresholds. Overall, the simulation demonstrates that PIPs provide an intuitive and more efficient indicator for feature selection than normalized posterior weights in SuSiE PCA. In addition, we also examined the sensitivity and specificity using weights estimated from sparse PCA and EBMF (see Figure S1), which have similar trends to the curves in Figure 1B and can only capture a small proportion of the true positive signals as the cutoff threshold increases.

SuSiE PCA is robust to model mis-specification

Next, we examined the estimation accuracy of the loading matrix as a function of sample size (*N*), feature dimension (*P*), latent dimension (*K*), and the number of single effects (or sparsity level) (*L*), via the Procrustes errors[16] (the Frobenius norm after Procrustes transformation,[17] see STAR Methods) (Figures 2A–2D). We found that SuSiE PCA has the smallest Procrustes errors across all simulation settings compared to sparse PCA and EBMF. And we noticed that the Bayesian methods including SuSiE PCA and EBMF maintain a low error even with a small sample size or high feature dimension. Moreover, we found that SuSiE PCA has the lowest relative root mean squared error (RRMSE) across all simulations compared with other methods (Figure S2); and EBMF and SuSiE PCA have a lower level of Procrustes error of factor **Z** than sparse PCA (Figure S3). In summary, SuSiE PCA exhibits the highest estimation accuracy, which is consistent with its superior performance in variable selection.

We next investigated model robustness under model mis-specification. Similar to other latent factor models, SuSiE PCA could be mis-specified as it requires manually inputting the latent dimension *K* and the number of single effects *L*. Considering the potential model mis-specification setting, the simulation datasets are generated based on $K = 4, L = 40$ and then input into SuSiE PCA, sparse PCA, and EBMF with two mis-specified situations: vary *L* while fixing *K*, or vary *K* while fixing *L*. The model estimation accuracy is then compared among three models with Procrustes error (see Figures 2E and 2F). We observed that as *K* and *L* in the model approach the true value (i.e., $K = 4$ or $L = 40$), the Procrustes error decreases rapidly to the lower level in SuSiE PCA and remains the same even when $K>4$ or $L>40$. However, the error for sparse PCA has a V shape and reaches its minimum at the real *K*. The explanation is that when there are over-specified latent factors in the model, SuSiE PCA and EBMF will not extract any information from the data due to their probabilistic model structure; the sparse PCA, on the other hand, cannot handle the weights since it does not impose a probabilistic assumption on them. Instead, the value of the redundant latent factor in sparse PCA is close to 0, which ensures the latent component does not contribute.

Finally, to compare the generative capacity, we computed and compared the log likelihood of held-out data between sparse PCA and SuSiE PCA. We observed that SuSiE PCA outperforms sparse PCA and obtains higher log likelihoods for simulations (Figure S5). In addition

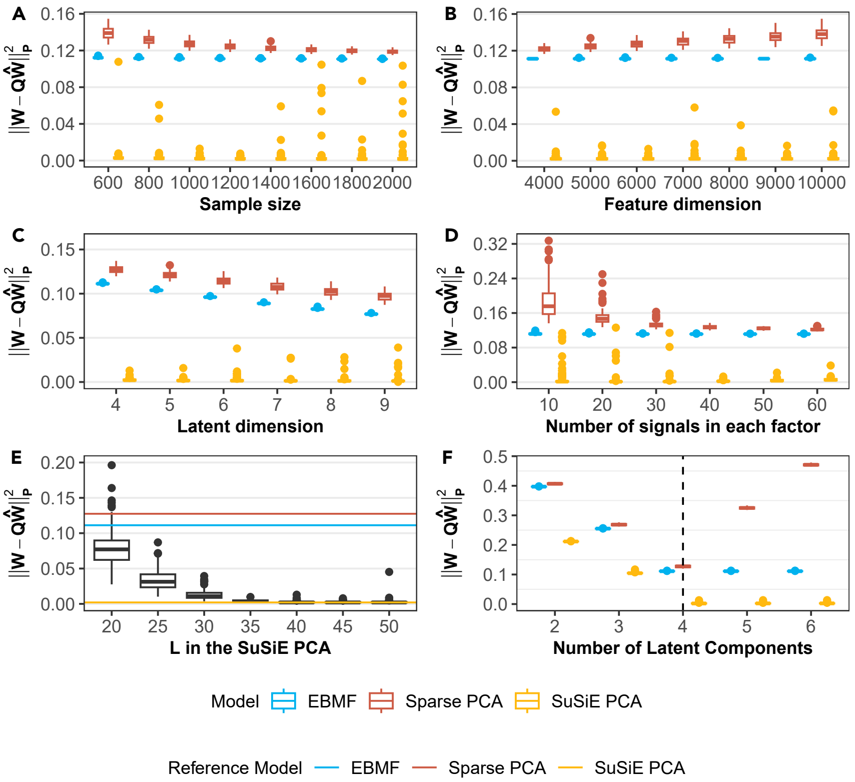


Figure 2. SuSiE PCA outperforms sparse PCA and EBMF in estimation accuracy and model robustness

SuSiE PCA generates the smallest Procrustes error in weight matrix than sparse PCA and EBMF (A–D) and is robust to over-specified K and L (E and F). For each scenario in (A–D) we vary one of the parameters at a time to generate the simulation data while fixing the other three parameters, and then input the true parameters (N, P, K, L) into models. Finally, we compute the Procrustes error and plot them as a function of N, P, K, L. For (E and F), we use the same simulation setting in Figure 1 to generate data but vary the specified L in SuSiE PCA (E) and K in all three models (F). Reference lines refer to the error from the models with correctly specified parameters (i.e., $L = 40, K = 4$).

to the overall superior model performance, SuSiE PCA remains faster on both CPU and GPU than sparse PCA and EBMF due to the efficient variational algorithm we implement (see STAR Methods) with the JAX library developed by Google.[18]

Dissecting cross-tissue eQTLs in GTEx

To illustrate the utility of SuSiE PCA to make inferences in biological data, we analyzed multi-tissue eQTL Z score results computed from GTEx v8[12,14] (see STAR Methods). Specifically, we sought to identify latent factors corresponding to tissue-specific and tissue-shared eQTLs similar to ref. 12. Overall, we found that 27 latent factors explained 53.1% of the variance in the data (see Figure S6). Although we set $L = 18$ across all factors, we found the number of tissues with PIP >0.9 is frequently lower than 18 in different factors (see Figure S9), which is due to inferred τ_{0kl} acting to "shut off" uninformative features. Indeed, we observed 30 out of 486 τ_{0kl} with estimates greater than e^{10} (see Figure S7) which effectively shrink the effect size of the corresponding single effect toward 0, driving the number of non-zero single effects in some factors smaller than specified L. We found this behavior also reflected in estimated level-0.9 credible sets, where 456 out of 486 contained a single tissue, and the remaining 30 credible sets contained at least two tissues.

To understand what each factor represents, we examined inferred PIPs (Figure S9) and posterior mean weights of each tissue across 27 factors (Figure S8). Here we present the results from factor $\mathbf{z}_1$ and $\mathbf{z}_3$ through the posterior weights (Figure 3; see Figure S8 for the remainder). We observed that the latent factor $\mathbf{z}_1$ with the second largest percentage of variance (PVE) demonstrates high absolute weights on most tissues except for the brain tissues, while the latent factor $\mathbf{z}_3$ has large weights almost exclusively on brain tissues. Moreover, we observed that brain tissue tends to appear as a group and has similar effects, implying the eQTLs in brain tissue are different from those in other tissue and those strong signals are specifically captured by the factor $\mathbf{z}_1$. For the rest of the factors, we noticed that factors with large PVE such as $\mathbf{z}_2, \mathbf{z}_4, \mathbf{z}_5$ tended to have large weights on multiple tissues; for example, factor $\mathbf{z}_2$ has large weights on esophagus and thyroid, suggesting the eQTLs signals are mostly shared across those tissues, while the factors with small PVE usually have large weights exclusively on one or a few tissues, for example, liver-specific component $\mathbf{z}_{12}$, lung-specific component $\mathbf{z}_{15}$, etc. The only exception is that the factor $\mathbf{z}_0$ with the largest PVE has an exclusively large weight only on the testis, implying the $\mathbf{z}_0$ captures the testis-specific eQTL signals. This is consistent with the investigation of the latent factor values of $\mathbf{z}_0$: the gene with the largest factor value in $\mathbf{z}_0$ is D-dopachrome tautomerase (DDT) (Figure S10), which is shown to

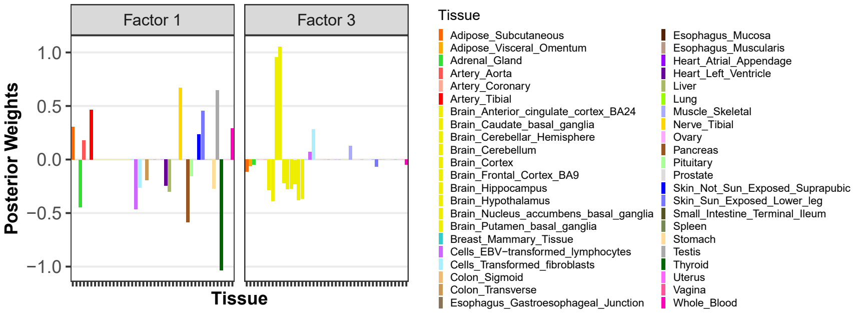


Figure 3. Factor $\mathbf{z}_1$ and $\mathbf{z}_3$ captures different types of tissues (tissues without brain vs. brain tissues)
The posterior weights refer to the inferred **W** matrix from the SuSiE PCA. The clustering pattern in different factors is found as there are only a limited number of tissues with non-zero weights in each factor since we set L = 18 while the feature dimension is 44.

be associated with testis cancer.[19] To make a comparison with the existing method, we expanded our investigation by applying sparse PCA to the GTEx Z score dataset and observed comparable tissue weights and factor scores across components in both SuSiE PCA and sparse PCA (Figure S11). However, a notable distinction arises where certain tissues exhibit tiny weights and can potentially be neglected in sparse PCA; in contrast, the SuSiE PCA can successfully capture the signals in those tissues through the PIP. For example, from the original analysis, both models identify adipose gland as the most relevant tissue in factor 10, while the remaining tissues have a much smaller relative weight and can effectively be ignored. Despite this, SuSiE PCA assigns a PIP of 1 to the lowly weighted tissues, suggesting that important signals would be missed if weights alone were used to provide insight. Overall, we find that SuSiE PCA is able to identify tissue-specific components from multi-tissue eQTL data in an intuitive, interpretable manner.

Identifying regulatory modules from perturb-seq data

To identify gene regulatory modules from genome-wide perturbation data, we ran SuSiE PCA on perturb-seq in cell lines[15] (see STAR Methods) with K = 10 and L = 300. Briefly, we inputted the normalized expression data (2057 × 8563) to SuSiE PCA to identify gene regulatory modules (i.e., **Z**) and downstream-regulated networks (i.e., **W**). To ensure our results were robust to K and L, we explored a grid of possible combinations and found that K = 10 and L = 300 retain the most important information while keeping the relevant gene set much smaller (see Figure S12 for a detailed explanation).

Overall, we found the total PVE was 10.71% across all components (Figure S13), with each component exhibiting 299 downstream genes with PIP $>$0.9 on average. Focusing on the leading component, we found that perturbations with the top 10 largest absolute factor scores are primarily related to Ribosomal Protein Small (RPS) subunit genes and Ribosomal Protein Large (RPL) subunit family (Figure 4A). To provide a broader characterization of the module function, we extracted downstream genes with PIP greater than 0.9 (298 genes) as input into ShinyGO[20] to perform a gene set enrichment analysis (Figure 4B). We observed the most enriched pathway was related to ribosome function (FDR = 9.2×10^{-82}, 63 genes involved), followed by coronavirus disease (FDR = 2.5×10^{-62}, 62 genes involved). Inspecting the loadings at these downstream genes, we found nearly all weights were positive, suggesting that the knockout of RPS and RPL genes down-regulates the expression level of those downstream genes. We found multiple elongation factor genes (EEF1G, EEF1A1, EEF1B2, EIF4B, EIF3L) among the leading downstream genes, which are known to be involved in ribosome function. Additionally, recent studies have suggested that the decreased expression of elongation factor genes is associated with less severe conditions among COVID-19 patients.[21,22] We repeated pathway analysis for each latent factor using corresponding loadings at genes with PIP greater than 0.9 (see Figures S14–S22).

To compare with sparse PCA, we performed the same pathway analysis on factor loadings and assessed enrichments. From the sparse PCA with the largest PVE (alpha = 1), we observed components identified by sparse PCA to be less enriched with biological pathways when compared to SuSiE PCA (80 unique enriched pathways in sparse PCA versus 88 pathways in SuSiE PCA), and the top enriched pathways such as ribosome and coronavirus disease are less significant and contain less number of selected genes (FDR = 1.4×10^{-33}, 35 genes; FDR = 2.9×10^{-18}, 29 genes). We noticed that, when alpha equals 17, the sparse PCA achieves an approximate similar total PVE (10.91%) with that of our model (10.71%) but with lower sparsity level (Figure S23). We then extracted the top 300 genes with non-zero weights in sparse PCA with alpha = 17 and performed the gene set enrichment analysis and found that the significance level is almost similar to that in SuSiE PCA (Figure S24). However, this is a post hoc analysis that suggests SuSiE PCA is more suitable for sparse data analysis while maintaining the power to perform the feature selection in a more statistical and reasonable manner.

Overall, we find distinct biological functions identified by each component, with groupings consistent with those reported in previous works.[23–25]

DISCUSSION

In this paper, we propose SuSiE PCA, an efficient Bayesian variable selection approach to PCA for structured biological data. The sparsity of the loading matrix is achieved by restricting the number of features associated with each factor to be at most L. Through simulations and

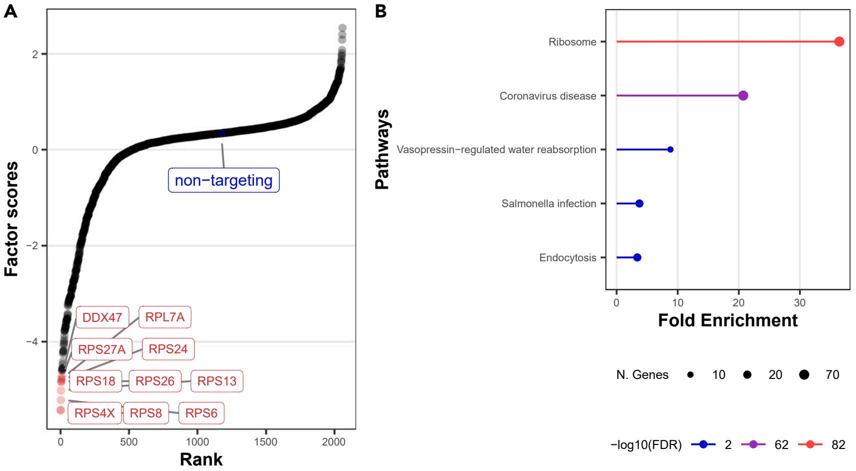


Figure 4. Dominant factor scores in top component link to RPL and RPS family with subsequent gene enrichment in Ribosome and Coronavirus disease

The perturbations with top factor scores in the first component mostly belong to RPL and RPS family(A), and the enrichment analysis results of downstream genes in the same component are enriched for ribosome and coronavirus disease(B). Each point in (A) represents the latent factor value of each perturbation. The top 9 points as well as the control group are labeled in the plot and colored red and blue, respectively. In gene set enrichment analysis, we input the downstream genes with PIP > 0.9 and show the top enriched pathways with log(FDR) and the number of genes included in the corresponding pathways.

real-data application, we find that SuSiE PCA outperforms existing approaches to sparse latent structure learning in identifying contributing features, while maintaining a more efficient run time.

There are several advantages of SuSiE PCA as compared to other sparse factor models. First, SuSiE PCA generates the PIPs for each feature that quantifies the uncertainty of the selected feature, which can not be provided by other sparse models, such as sparse PCA with regularization[13] or the Bayesian treatment of PPCA. And assessing the selected variables based on the probability is more reasonable and convenient than using weights. Second, PIPs are capable of selecting more signals with high confidence. In simulations, we demonstrated that using weights for variable selection from SuSiE PCA, sparse PCA, and EBMF can deliver a high specificity (low FDR) but with low sensitivity as the cutoff value increases, while using PIPs as selection tools can maintain a high sensitivity for any positive cutoff value between 0 and 1. Third, SuSiE PCA provides a more precise estimate of the loadings and higher prediction accuracy, even in the mis-specified case, as we impose a probabilistic distribution over the loadings that enables a much more accurate inference on the posterior distribution. Finally, the inference procedure of SuSiE PCA works on the dimension of K and L, which is typically set to be much smaller than feature dimension P; therefore, it is scalable to high-dimensional data and requires less computational demands. We implement the SuSiE PCA with the JAX library developed by Google[18] to enable fast convergence on CPU, GPU, or TPU. The comparison of run time among SuSiE PCA, sparse PCA, and EBMF is listed in Table 1.

In the SuSiE PCA, two parameters, the number of components K, and the number of single effects L, need to be prespecified by the user before fitting the model. The selection of K follows a similar strategy as conventional PCA, often informed by researchers' domain expertise. The merit of SuSiE PCA is that when there are excessive latent components being specified, the variance explained for those components would be extremely minimal with a near-zero count of single effects exhibiting PIP > 0.9. This effectively allows for an initial choice of a relatively large K and subsequently inspecting the PVE and PIPs in each component to decide the most suitable K.

The choice of L determines the sparsity in the SuSiE PCA. Although SuSiE PCA only allows one common L specified across all factors, the number of non-zero effects captured across factors can be varied and learned from the data. This is because we treat the inverse of variance τ_{0kl} of the l_{th} single effect in factor $\mathbf{z}_k$ as a random variable. As the Algorithm 1 demonstrates, the maximum likelihood estimate (MLE) of τ_{0kl} at the step 3 is derived before inference of other parameters. When the L specified in the model, for a certain factor k, is greater than the true number of signals associated with that factor, the MLE of the τ_{0kl} will be extremely large for those excessive single effects, which then shrinks

Table 1. Comparison of mean and standard deviation of running time (seconds) between models

Model[a]	Simulation[b]	GTEx *Z* score	Perturb-seq
SuSiE PCA	3.14(0.49)	1.20	68.11
Sparse PCA	51.96(33.50)	41.22	1213.21
EBMF	39.83(5.80)	498.60	243.03

[a] All run time data in the table are based on the analyses performed on the same CPU for consistency. The CPU we used is the Apple M2 chip with 16 GB memory.
[b] Run time for simulation is recorded based on simulation setting in Figure 1, i.e., $N = 1000, P = 6000, K = 4, L = 40$; the average run time and corresponding standard deviation are computed for 100 simulations. We presented a more detailed run time comparison in simulation in Figure S4.

Algorithm 1. Algorithm for SuSiE PCA

```
Require: Data $\mathrm{X}_{N \times P}$
    Require: Number of Factors $K$; Number of single effects in each factor $L$
    Require: Initialize variational parameters $(\boldsymbol{\mu}_{\mathrm{Z}}, \boldsymbol{\Sigma}_{\mathrm{Z}}; \boldsymbol{\mu}_{\mathrm{w}_{kl}}, \boldsymbol{\sigma}_{\mathrm{w}_{kl}}; \boldsymbol{\alpha}_{kl})$; hyperparameters $\tau, \tau_{0kl}$, for $l = 1, \cdots, L; k = 1, \cdots, K$
    Require: update equations on different variables $F_{\mathrm{Z}}; F_{\mathrm{w}_{kl}}; F_{\boldsymbol{\alpha}_{kl}}; F_{\boldsymbol{\tau}_0}; F_{\tau}$:
    Require: function to compute ELBO, $F_{\mathrm{ELBO}}$
    Ensure: ELBO increase
    repeat
        $\mathrm{W} \leftarrow \sum_{l=1}^{L} \boldsymbol{\mu}_{\mathrm{w}} \circ \boldsymbol{\alpha}$. ▷ Define $\boldsymbol{\mu}_{\mathrm{w}}, \boldsymbol{\alpha}$ as $(L, K, P)$ arrays by arranging $\boldsymbol{\mu}_{\mathrm{w}_{kl}}, \boldsymbol{\alpha}_{kl}$
        $\boldsymbol{\tau}_0 \leftarrow F_{\boldsymbol{\tau}_0}(\boldsymbol{\mu}_{\mathrm{w}}, \boldsymbol{\sigma}_{\mathrm{w}}, \boldsymbol{\alpha})$
        for $k$ in $1, \cdots, K$ do
            $\mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k]^{(1)} = \mathrm{X}^{\top} \boldsymbol{\mu}_{\mathrm{z}_k} - \sum_{k' \neq k} \mathbb{E}[\mathrm{w}_{k'}] \mathbb{E}[\mathrm{Z}_{k'}^{\top} \mathrm{Z}_k]$ ▷ compute the first two terms in Eq
            for $l$ in $1, \cdots, L$ do
                $\mathbb{E}[\mathrm{w}_{kl'}] = \mathrm{w}_k - \boldsymbol{\mu}_{\mathrm{w}_{kl}} \circ \boldsymbol{\alpha}_{kl}$ ▷ removing the $l_{th}$ effect from $\mathrm{w}_k$
                $\mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k] = \mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k]^{(1)} - \mathrm{w}_k \mathbb{E}[\mathrm{Z}_k^{\top} \mathrm{Z}_k]$ ▷ complete the calculation of $\mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k]$
                $(\boldsymbol{\mu}_{\mathrm{w}_{kl}}, \boldsymbol{\sigma}_{\mathrm{w}_{kl}}) \leftarrow F_{\mathrm{w}_{kl}}(\mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k], \mathbb{E}[\mathrm{Z}_k^{\top} \mathrm{Z}_k], \tau_{0kl}, \boldsymbol{\tau})$
                $\boldsymbol{\alpha}_{kl} \leftarrow F_{\boldsymbol{\alpha}_{kl}}(\mathbb{E}[\mathrm{R}_{kl}^{\top} \mathrm{Z}_k], \boldsymbol{\mu}_{\mathrm{w}_{kl}}, \boldsymbol{\sigma}_{\mathrm{w}_{kl}})$
                $\mathrm{w}_k = \mathbb{E}[\mathrm{w}_{kl'}] + \mu_{\mathrm{w}_{kl}} \circ \boldsymbol{\alpha}_{kl}$ ▷ Update the $\mathrm{w}_k$
            end for
        end for
        $(\boldsymbol{\mu}_{\mathrm{Z}}, \boldsymbol{\Sigma}_{\mathrm{Z}}) \leftarrow F_Z(\mathrm{X}, \boldsymbol{\tau}, \mathbb{E}[\mathrm{W}])$
        $\boldsymbol{\tau} = F_{\tau}(\mathrm{X}, \boldsymbol{\tau}, \mathbb{E}[\mathrm{W}], \mathbb{E}[\mathrm{Z}])$
    $ELBO \leftarrow F_{\mathrm{ELBO}}$
    until ELBO convergence criterion satisfied
```

the $\mathbf{w}_{kl}$ and PIP to be 0 or close to 0, and therefore removes the redundant single effects from the model. For example, in the simulation and GTEx Z score data analysis, we have shown that when the user-specified L is larger than the data-generating L, the automatic relevance determination-like (ARD) prior over loadings will shrink effects toward 0, thus adding little additional predictive power and overall mean square error (MSE) from the true loadings matrix. Although it seems like the L parameter may be automatically set to the total number of variables (and thus "shut off" if necessary), we emphasize that this still comes with an added computational cost, albeit a low one due to the scalability of our approach. Therefore, we allow users to specify their own choice of L. From this point of view, without prior knowledge of the data, one can specify a relatively larger L during the initial model fitting and then examine the estimates of τ_{0kl} to explore how many single effects are reasonable for the dataset.

Overall, SuSiE PCA provides a flexible approach to high-dimensional biological data with a low-rank structure and allows for feature selection in sparse PCA.

Limitations of the study

One limitation of SuSiE PCA is that under the mean-field approximation, all the posteriors, i.e., $Q(\mathbf{W})$ and $Q(\mathbf{Z})$, are factorized to facilitate inference. Under this factorization, estimation for mean terms (i.e., $E[\mathbf{W}]$ and $E[\mathbf{Z}]$) is approximately unbiased.[26] However, it produces overconfident covariance structures within variables ($\mathbf{W}$, $\mathbf{Z}$, etc) due to the assumed independence across Q functions.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
    - ○ Lead contact
    - ○ Material availability
    - ○ Data and code availability
- EXPERIMENTAL MODEL AND SUBJECT DETAILS
- METHOD DETAILS
    - ○ Overview of SuSiE PCA
    - ○ Posterior inclusion probability
    - ○ Variational inference in SuSiE PCA
    - ○ Mean-field approximation

- ○ Helpful definitions
- ○ Derivation of model parameters
- ○ Derivation of evidence lower bound (ELBO)
- ○ Simulations
- ○ GTEx *Z* score dataset
- ○ Purturb-seq dataset

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2023.108181.

ACKNOWLEDGMENTS

This work was funded by the National Institutes of Health (NIH) under awards R01HG012133, P01CA196569.

AUTHOR CONTRIBUTIONS

D.Y. and N.M. developed the method. D.Y. performed analysis. D.Y. and N.M. edited and approved the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 29, 2023
Revised: September 20, 2023
Accepted: October 9, 2023
Published: October 13, 2023

REFERENCES

1. Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. J. Educ. Psychol. *24*, 417–441. https://doi.org/10.1037/h0071325.
2. Patterson, N., Price, A.L., and Reich, D. (2006). Population Structure and Eigenanalysis. PLoS Genet. *2*, e190. https://doi.org/10.1371/journal.pgen.0020190.
3. Agrawal, A., Chiu, A.M., Le, M., Halperin, E., and Sankararaman, S. (2020). Scalable probabilistic PCA for large-scale genetic variation data. PLoS Genet. *16*, e1008773. https://doi.org/10.1371/journal.pgen.1008773.
4. McVean, G. (2009). A Genealogical Interpretation of Principal Components Analysis. PLoS Genet. *5*, e1000686. https://doi.org/10.1371/journal.pgen.1000686.
5. Jolliffe, I.T. (1986). Principal Component Analysis (Springer-Verlag).
6. Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse Principal Component Analysis. J. Comput. Graph Stat. *15*, 265–286. https://doi.org/10.1198/106186006X113430.
7. Bishop, C. (1998). Bayesian PCA. In Advances in Neural Information Processing Systems (MIT Press).
8. Guan, Y., and Dy, J. (2009). Sparse Probabilistic Principal Component Analysis. In Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics (PMLR), pp. 185–192.
9. Ning, B. (2021). Spike and slab Bayesian sparse principal component analysis. Preprint at arXiv. https://doi.org/10.48550/arXiv.2102.00305.
10. Armagan, A., Clyde, M., and Dunson, D. (2011). Generalized Beta Mixtures of Gaussians. In Advances in Neural Information Processing Systems (Curran Associates, Inc.).
11. Zhao, S., Gao, C., Mukherjee, S., and Engelhardt, B.E. (2016). Bayesian group factor analysis with structured sparsity. J. Mach. Learn. Res. *17*, 1–47.
12. Wang, W., Ge, L., Zhang, L., Liu, L., Zhang, X., and Ma, X. (2021). Empirical bayes matrix factorization. Hum. Fertil. *22*, 1–11.
13. Wang, G., Sarkar, A., Carbonetto, P., and Stephens, M. (2020). A simple new approach to variable selection in regression, with application to genetic fine mapping. J. R. Stat. Soc. Series B Stat. Methodol. *82*, 1273–1300. https://doi.org/10.1111/rssb.12388.
14. GTEx Consortium, Ardlie, K.G., Deluca, D.S., Segrè, A.V., Sullivan, T.J., Young, T.R., Gelfand, E.T., Trowbridge, C.A., Maller, J.B., Tukiainen, T., et al. (2015). The Genotype-Tissue Expression (GTEx) pilot analysis: Multitissue gene regulation in humans. Science *348*, 648–660. https://doi.org/10.1126/science.1262110.
15. Replogle, J.M., Saunders, R.A., Pogson, A.N., Hussmann, J.A., Lenail, A., Guna, A., Mascibroda, L., Wagner, E.J., Adelman, K., Lithwick-Yanai, G., et al. (2022). Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq. Cell *185*, 2559–2575.e28. https://doi.org/10.1016/j.cell.2022.05.013.
16. Meng, F., Richer, M., Tehrani, A., La, J., Kim, T.D., Ayers, P.W., and Heidar-Zadeh, F. (2022). Procrustes: A python library to find transformations that maximize the similarity between matrices. Comput. Phys. Commun. *276*, 108334. https://doi.org/10.1016/j.cpc.2022.108334.
17. Borg, I., and Groenen, P. (2005). Modern Multidimensional Scaling: Theory and Applications. Springer Series in Statistics). https://doi.org/10.1007/978-1-4757-2711-1.
18. Bradbury, J., Frostig, R., Hawkins, P., Johnson, M.J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., et al. (2018). JAX: Composable Transformations of Python+NumPy Programs.
19. Cohn, B.A., Cirillo, P.M., and Christianson, R.E. (2010). Prenatal DDT Exposure and Testicular Cancer: A Nested Case-Control Study. Arch. Environ. Occup. Health *65*, 127–134. https://doi.org/10.1080/19338241003730887.
20. Ge, S.X., Jung, D., and Yao, R. (2020). ShinyGO: a graphical gene-set enrichment tool for animals and plants. Bioinformatics *36*, 2628–2629. https://doi.org/10.1093/bioinformatics/btz931.
21. Amrute, J.M., Perry, A.M., Anand, G., Cruchaga, C., Hock, K.G., Farnsworth, C.W., Randolph, G.J., Lavine, K.J., and Steed, A.L. (2022). Cell specific peripheral immune responses predict survival in critical COVID-19 patients. Nat. Commun. *13*, 882. https://doi.org/10.1038/s41467-022-28505-3.
22. Garg, M., Li, X., Moreno, P., Papatheodorou, I., Shu, Y., Brazma, A., and Miao, Z. (2021). Meta-analysis of COVID-19 single-cell studies confirms eight key immune responses. Sci. Rep. *11*, 20833. https://doi.org/10.1038/s41598-021-00121-z.
23. Signorile, A., Sgaramella, G., Bellomo, F., and De Rasmo, D. (2019). Prohibitins: A Critical Role in Mitochondrial Functions and Implication in Diseases. Cells *8*, 71. https://doi.org/10.3390/cells8010071.
24. Artal-Sanz, M., Tsang, W.Y., Willems, E.M., Grivell, L.A., Lemire, B.D., van der Spek, H., and Nijtmans, L.G.J. (2003). The

mitochondrial prohibitin complex is essential for embryonic viability and germline function in Caenorhabditis elegans. J. Biol. Chem. *278*, 32091–32099. https://doi.org/10.1074/jbc.M304877200.

25. Artal-Sanz, M., and Tavernarakis, N. (2009). Prohibitin couples diapause signalling to mitochondrial metabolism during ageing in C. elegans. Nature *461*, 793–797. https://doi.org/10.1038/nature08466.

26. Opper, M., and Saad, D. (2001). Advanced Mean Field Methods: Theory and Practice (MIT Press).

27. Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M.I. (2003). An Introduction to MCMC for Machine Learning. Mach. Learn. *50*, 5–43. https://doi.org/10.1023/A:1020281327116.

28. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., and Saul, L.K. (1999). An Introduction to Variational Methods for Graphical Models. Mach. Learn. *37*, 183–233. https://doi.org/10.1023/A:1007665907178.

29. Kullback, S., and Leibler, R.A. (1951). On Information and Sufficiency. Ann. Math. Stat. *22*, 79–86.

30. Tanaka, T. (1998). A Theory of Mean Field Approximation. In Advances in Neural Information Processing Systems (MIT Press).

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
The Genotype-Tissue Expression *Z* score data	Wei Wang and Matthew Stephens, Empirical Bayes Matrix Factorization, 2021[1]	https://github.com/ysfoo/sparsefactor
Genome-scale Perturb-seq experiment data	Joseph Replogle and Jonathan Weissman, Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq, 2022[2]	https://plus.figshare.com/articles/dataset/_Mapping_information-rich_genotype-phenotype_landscapes_with_genome-scale_Perturb-seq_Replogle_et_al_2022_processed_Perturb-seq_datasets/20029387
Software and algorithms		
Scikit-learn library: sparse principal component analysis	Python library scikit-learn	https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.SparsePCA.html; RRID:SCR_002577
R Package: Factors and Loadings by Adaptive SHrinkage in R (flashr)	Wei Wang and Matthew Stephens, Empirical Bayes Matrix Factorization[1]	https://stephenslab.github.io/flashr/index.html
Variational algorithm in SuSiE PCA	This paper	http://www.github.com/mancusolab/susiepca
Python 3.9	Python Software Foundation	https://www.python.org/
R 4.0.0	R Software	https://www.r-project.org
ShinyGO v0.77	Ge SX, Jung D & Yao R[3]	http://bioinformatics.sdstate.edu/go/; RRID:SCR_019213

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Dong Yuan (dongyuan@usc.edu).

Material availability

This study did not generate new unique materials or reagents.

Data and code availability

- This paper analyzes existing, publicly available data, i.e., the GTEx z-score dataset[14] and the perturb-seq data.[15] These accession numbers for the datasets are listed in the key resources table.
- All original codes related to SuSiE PCA have been deposited and are publicly available on GitHub (https://github.com/mancusolab/susiepca).
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

EXPERIMENTAL MODEL AND SUBJECT DETAILS

This study did not include experiments with a specific model or subject.

METHOD DETAILS

Overview of SuSiE PCA

In this section, we will give a detailed description of SuSiE PCA. Let $\mathbf{X}_{N\times P}$ be the observed data matrix, $\mathbf{Z}_{N\times K}$ be the K dimensional latent vectors, and $\mathbf{W}_{K\times P}$ be the loading matrix. We denote the normal distribution with mean μ and variance σ^2 as $\mathcal{N}(\mu,\sigma^2)$, the multinomial distribution with n choices and probabilities $\boldsymbol{\pi}$ as $\text{Multi}(n,\boldsymbol{\pi})$ and the matrix normal distribution with dimension $N\times K$, mean $\mathbf{M}$, row-covariance $\mathbf{R}$, and column-covariance $\mathbf{C}$ as $\mathcal{MN}_{N,K}(\mathbf{M},\mathbf{R},\mathbf{C})$. We denote the basis vector in which k^{th} coordinate is 1 and 0 elsewhere as $\mathbf{e}_k$. The sampling distribution of $\mathbf{X}$ under the SuSiE PCA model is given by,

$$\mathbf{X} \mid \mathbf{Z}, \mathbf{W}, \sigma^2 \sim \mathcal{MN}_{N,P}\left(\mathbf{ZW}, \mathbf{I}_N, \sigma^2\mathbf{I}_P\right) \qquad \text{(Equation 1)}$$

$$\mathbf{Z} \sim \mathcal{MN}_{N,K}(0, \mathbf{I}_N, \mathbf{I}_K) \qquad \text{(Equation 2)}$$

$$\mathbf{W} = \sum_{k=1}^{K} \mathbf{e}_k \mathbf{w}_k^{\top} \qquad \text{(Equation 3)}$$

$$\mathbf{w}_k = \sum_{l=1}^{L} \mathbf{w}_{kl} \qquad \text{(Equation 4)}$$

$$\mathbf{w}_{kl} = w_{kl} \boldsymbol{\gamma}_{kl} \qquad \text{(Equation 5)}$$

$$w_{kl} \mid \sigma_{0kl}^2 \sim \mathcal{N}(0, \sigma_{0kl}^2) \qquad \text{(Equation 6)}$$

$$\boldsymbol{\gamma}_{kl} \mid \boldsymbol{\pi} \sim \text{Multi}(1, \boldsymbol{\pi}), \qquad \text{(Equation 7)}$$

where $\mathbf{w}_k$ corresponds to the k^{th} row of $\mathbf{W}$, and contains at most L non-zero elements determined by the sum of L single-effect vectors $\mathbf{w}_{kl}$. These single-effect vectors are described by a single random effect w_{kl} and indicator vector $\boldsymbol{\gamma}_{kl}$ which assigns the effect to a feature with prior probabilities $\boldsymbol{\pi} = \frac{1}{p}\mathbf{1}$.

Posterior inclusion probability

One of the distinguishing features that the SuSiE model[13] provides is a posterior inclusion probability (PIP). The PIP reflects the posterior probability that a given variable has a non-zero effect given the observed data. Here we extend the PIP definition to include latent factors. Specifically, given variational parameters $\boldsymbol{\alpha}_{kl}$ we can define the PIP that the i^{th} variable has a non-zero effect in the k^{th} latent component as,

$$\text{PIP}_{ki} := \Pr(w_{ki} \neq 0 \mid \mathbf{X}) = 1 - \prod_{l=1}^{L} (1 - \alpha_{kli}) \qquad \text{(Equation 8)}$$

Similarly, a level-ρ credible set (CS) refers to a subset of variables that cumulatively explain at least ρ of the posterior density. Here, we define factor-specific level-ρ CSs, which can be computed across each $\boldsymbol{\alpha}_{kl}$ independently, resulting in $K \times L$ total level-ρ credible sets. This lets us reflect on the uncertainty in identified variables to explain a single-effect for each latent factor.

Variational inference in SuSiE PCA

We seek to perform inference of model variables $\mathbf{Z}$, $\mathbf{w}_{kl}$ and $\boldsymbol{\gamma}_{kl}$ conditional on observed data $\mathbf{X}$, however, the marginal likelihood is intractable to compute and therefore, we cannot evaluate the posterior exactly. While sampling based approaches such as Markov Chain Monte Carlo (MCMC) methods provide a numerical approximation of the exact posterior distribution,[27] they often lack computational efficiency in high-dimensional settings. As an alternative, we leverage recent advancements in the variational inference that provides an analytical approximation to the posterior distribution[28] and remains computationally efficient.

Briefly, To approximate the conditional distribution of latent variables $\mathbf{Z}$ given the observed samples $\mathbf{X}$, variational methods first impose a family of densities over the latent variables, $Q(\mathbf{Z})$, usually predefined as known distributions parameterized with a set of variational parameters. Then the goal is to infer those variational parameters such that the variational distribution $Q(\mathbf{Z})$ is as similar as possible to the true posterior distribution $P(\mathbf{Z} \mid \mathbf{X})$. A quantity commonly used to measure dissimilarity between distributions is Kullback-Leibler divergence $D_{KL}(Q \parallel P)$.[29] However, since KL divergence contain the unknown true posterior distribution $P(\mathbf{Z} \mid \mathbf{X})$, it cannot be directly computed. Instead, we can show that the log-likelihood of data, $\log P(\mathbf{X})$ can be decomposed as:

$$\log P(\mathbf{X}) = D_{KL}(Q \parallel P) + \mathcal{L}(Q) \qquad \text{(Equation 9)}$$

Where $\mathcal{L}(Q) = \mathbb{E}_Q[\log P(\mathbf{Z},\mathbf{X}) - \log Q(\mathbf{Z})]$, which is also known as the Evidence Lower Bound (ELBO). Since the $\log P(\mathbf{X})$ is a constant with respect to the variational parameters, minimizing KL divergence is equivalent to maximizing ELBO. As the ELBO does not contain the unknown posterior distribution and therefore is tractable to compute and maximize for variational parameters.

Mean-field approximation

Mean field approximation[30] is a common solution to find the optimal solution that maximizes ELBO. The basic assumption is that we can factorize the variational distribution into independent components. Then using the calculus of variations, one can show that the distribution $Q_j^*(\mathbf{z}_j)$ minimizing KL divergence for each factor $\mathbf{Z}_j$ can be expressed as:

$$\ln Q_j^*(\mathbf{z}_j \mid \mathbf{X}) = \mathbb{E}_{i \neq j}[\ln P(\mathbf{Z}, \mathbf{X})] + \textit{constant} \qquad \text{(Equation 10)}$$

Applying the Mean-Field approximation to SuSiE PCA the approximate posterior given by,

$$Q(\mathrm{Z}, \mathrm{W}) = Q(\mathrm{Z})Q(\mathrm{W}) \qquad \text{(Equation 11)}$$

$$Q(\mathrm{W}) = \prod_{k=1}^{K} \prod_{l=1}^{L} Q(\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}) Q(\boldsymbol{\gamma}_{kl}) \qquad \text{(Equation 12)}$$

Equation 11 factorizes the variational densities of the latent variables **Z** and the loading matrix **W** into independent parts. We further assume that the variational distribution of loadings $\mathbf{w}_{kl}$ from each factor across L single effects are independent as well, leading to Equation 12. For ease of notation we first define $\tau = \frac{1}{\sigma^2}, \tau_{0kl} = \frac{1}{\sigma^2_{0kl}}$. Based on the factorization, the complete-data log-likelihood of data and parameters of SuSiE PCA is given by:

$$\begin{aligned} m_c\left(\sigma^2, \sigma_0^2, \boldsymbol{\pi} \,\middle|\, \mathrm{X}, \mathrm{Z}, \mathrm{W}\right) &= \log\Pr\left(\mathrm{X} \,\middle|\, \mathrm{Z}, \mathrm{W}, \sigma^2\right) + \log\Pr(\mathrm{Z}) + \log\Pr\left(\mathrm{W} \,\middle|\, \sigma_0^2, \boldsymbol{\pi}\right) \\ &= \log \mathcal{MN}_{n,p}\left(\mathrm{X} \,\middle|\, \mathrm{ZW}, \mathrm{I}_n, \mathrm{I}_p\sigma^2\right) + \log \mathcal{MN}_{n,k}(\mathrm{Z} \mid 0, \mathrm{I}_n, \mathrm{I}_k) + \\ &\sum_{l=1}^{L} \sum_{k=1}^{K} \left[\log\mathrm{Multi}(\boldsymbol{\gamma}_{kl} \mid 1, \boldsymbol{\pi}) + \log \mathcal{N}\left(w_{kl} \,\middle|\, 0, \sigma_0^2\right)\right] \end{aligned}$$

Helpful definitions

Before proceeding to the full derivation of variational distribution of parameters **Z**,$\mathbf{w}_{kl}$,and $\boldsymbol{\gamma}_{kl}$, we first give some helpful definitions, including the expansion of the first and second moment of **W** and **Z**.

The second moment of **Z** is:

$$\begin{aligned} \mathbb{E}[\mathrm{Z}^\top \mathrm{Z}] &= \mathrm{tr}(\mathrm{I}_n)\boldsymbol{\Sigma}_{\mathrm{Z}} + \mathbb{E}[\mathrm{Z}]^\top \mathbb{E}[\mathrm{Z}] \\ &= n\boldsymbol{\Sigma}_{\mathrm{Z}} + \mathbb{E}[\mathrm{Z}]^\top \mathbb{E}[\mathrm{Z}] \end{aligned}$$

$$\begin{aligned} \mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right] &= \mathrm{tr}(\mathbb{V}[\mathrm{Z}_k]) + \mathbb{E}[\mathrm{Z}_k]^\top \mathbb{E}[\mathrm{Z}_k] \\ &= \mathrm{tr}\left(\mathrm{I}_n(\boldsymbol{\Sigma}_{\mathrm{Z}})_{kk}\right) + \mathbb{E}[\mathrm{Z}_k]^\top \mathbb{E}[\mathrm{Z}_k] \\ &= n(\boldsymbol{\Sigma}_{\mathrm{Z}})_{kk} + \mathbb{E}[\mathrm{Z}_k]^\top \mathbb{E}[\mathrm{Z}_k] \end{aligned}$$

The first and second moments of $\mathbf{w}_k$ are listed as follows:

$$\mathbb{E}[\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] = \mathrm{p} - \text{vector of posterior}$$

conditional means

$$\mathbb{V}[\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] = \mathrm{p} - \text{vector of posterior}$$

conditional variances

$$\mathbb{E}[\mathbf{w}_k] = \mathbb{E}\left[\sum_l \mathbf{w}_{kl}\right] = \sum_l \mathbb{E}[\mathbf{w}_{kl}]$$

$$\mathbb{E}[\mathbf{w}_{kl}] = \sum_l \mathbb{E}[\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] \circ \mathbb{E}[\boldsymbol{\gamma}_{kl}]$$

$$\mathbb{V}[\mathbf{w}_k] = \mathbb{V}\left[\sum_l \mathbf{w}_{kl}\right] = \sum_l \mathbb{V}[\mathbf{w}_{kl}]$$

$$\begin{aligned} \mathbb{V}[\mathbf{w}_{kl}] &= \mathbb{E}\left[\mathbf{w}_{kl}\mathbf{w}_{kl}^\top\right] - \mathbb{E}[\mathbf{w}_{kl}]\mathbb{E}[\mathbf{w}_{kl}]^\top \\ &= \mathbb{E}\left[w_{kl}^2 \boldsymbol{\gamma}_{kl} \boldsymbol{\gamma}_{kl}^\top\right] - \mathbb{E}[\mathbf{w}_{kl}]\mathbb{E}[\mathbf{w}_{kl}]^\top \\ &= \mathrm{diag}(\mathbb{E}[\mathbf{w}_{kl} \circ \mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] \circ \mathbb{E}[\boldsymbol{\gamma}_{kl}]) - \mathbb{E}[\mathbf{w}_{kl}]\mathbb{E}[\mathbf{w}_{kl}]^\top \end{aligned}$$

$$\mathrm{diag}(\mathbb{V}[\mathbf{w}_{kl}]) = \mathbb{E}[\mathbf{w}_{kl} \circ \mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] \circ \mathbb{E}[\boldsymbol{\gamma}_{kl}] - (\mathbb{E}[\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kl}] \circ \mathbb{E}[\boldsymbol{\gamma}_{kl}])^2$$

$$\mathbb{E}\left[\mathbf{w}_k^\top \mathbf{w}_k\right] = \mathrm{tr}(\mathbb{V}[\mathbf{w}_k]) + \mathbb{E}[\mathbf{w}_k]^\top \mathbb{E}[\mathbf{w}_k]$$

$$\mathbb{E}\left[w_{kl}^2\right] = \left[\mathbb{E}^2[w_{kl} \mid \boldsymbol{\gamma}_{kl}] + \mathbb{V}[w_{kl} \mid \boldsymbol{\gamma}_{kl}]\right] \circ \mathbb{E}[\boldsymbol{\gamma}_{kl}]$$

The first and second moments of **W** are listed as follows:

$$\mathbb{E}[\mathbf{W}] = \mathbb{E}\left[\sum_k \mathbf{e}_k \mathbf{w}_k^\top\right] = \sum_k \mathbf{e}_k \mathbb{E}[\mathbf{w}_k]^\top$$

$$\begin{aligned}
\mathbb{E}[\mathbf{W}\mathbf{W}^\top] &= \mathbb{E}\left[\left(\sum_k \mathbf{e}_k \mathbf{w}_k^\top\right)\left(\sum_{k'} \mathbf{e}_{k'} \mathbf{w}_{k'}^\top\right)^\top\right] \\
&= \mathbb{E}\left[\sum_k\sum_{k'} \mathbf{e}_k \mathbf{w}_k^\top \mathbf{w}_{k'} \mathbf{e}_{k'}^\top\right] \\
&= \sum_k\sum_{k'} \mathbf{e}_k \mathbf{e}_{k'}^\top \mathbb{E}\left[\mathbf{w}_k^\top \mathbf{w}_{k'}\right] \\
&= \sum_k\sum_{k'} \mathbf{e}_k \mathbf{e}_{k'}^\top \mathbb{E}[\mathbf{w}_k]^\top \mathbb{E}[\mathbf{w}_{k'}] + \sum_k \mathbf{e}_k \mathbf{e}_k^\top \left(\mathbb{E}\left[\mathbf{w}_k^\top \mathbf{w}_k\right] - \mathbb{E}[\mathbf{w}_k]^\top \mathbb{E}[\mathbf{w}_k]\right) \\
&= \mathbb{E}[\mathbf{W}]\mathbb{E}[\mathbf{W}]^\top + \sum_k \mathbf{e}_k \mathbf{e}_k^\top \left(\mathbb{E}\left[\mathbf{w}_k^\top \mathbf{w}_k\right] - \mathbb{E}[\mathbf{w}_k]^\top \mathbb{E}[\mathbf{w}_k]\right) \\
&= \mathbb{E}[\mathbf{W}]\mathbb{E}[\mathbf{W}]^\top + \sum_k \mathbf{e}_k \mathbf{e}_k^\top \mathrm{tr}(\mathbb{V}[\mathbf{w}_k]) \\
&= \mathbb{E}[\mathbf{W}]\mathbb{E}[\mathbf{W}]^\top + \mathrm{diag}(\mathrm{tr}(\mathbb{V}[\mathbf{w}_1]), \ldots, \mathrm{tr}(\mathbb{V}[\mathbf{w}_k]))
\end{aligned}$$

Other terms in likelihood function:

$$\log \mathcal{MN}_{n,p}\left(\mathbf{X} \mid \mathbf{ZW}, \mathbf{I}_n, \mathbf{I}_p\sigma^2\right) = -\frac{1}{2\sigma^2}\mathrm{tr}[(\mathbf{X} - \mathbf{ZW})^\top(\mathbf{X} - \mathbf{ZW})] - \frac{np}{2}\log\left(2\pi\sigma^2\right)$$

$$\log \mathcal{MN}_{n,k}(\mathbf{Z} \mid 0, \mathbf{I}_n, \mathbf{I}_k) = -\frac{1}{2}\mathrm{tr}[\mathbf{Z}^\top\mathbf{Z}] - \frac{nk}{2}\log(2\pi)$$

$$\log\mathrm{Multi}(\boldsymbol{\gamma}_{kl} \mid 1, \boldsymbol{\pi}) = \sum_{i=1}^{p} \boldsymbol{\gamma}_{kli} \log(\pi_i)$$

$$\log \mathcal{N}\left(w_{kl} \mid 0, \sigma_0^2\right) = -\frac{1}{2\sigma_0^2}w_{kl}^2 - \frac{1}{2}\log\left(2\pi\sigma_0^2\right)$$

$$\begin{aligned}
\mathrm{tr}[\mathbb{E}_{\neg\mathbf{Z}}[(\mathbf{X} - \mathbf{ZW})^\top(\mathbf{X} - \mathbf{ZW})]] &= \mathrm{tr}[\mathbb{E}_{\neg\mathbf{Z}}(\mathbf{X}^\top\mathbf{X} - \mathbf{X}^\top\mathbf{ZW} - \mathbf{W}^\top\mathbf{Z}^\top\mathbf{X} + \mathbf{W}^\top\mathbf{Z}^\top\mathbf{ZW})] \\
&= \mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbb{E}[\mathbf{W}]\mathbf{X}^\top\mathbf{Z}) + \mathrm{tr}(\mathbf{Z}^\top\mathbf{Z}\mathbb{E}[\mathbf{W}\mathbf{W}^\top]) \\
&= \mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbb{E}[\mathbf{W}]\mathbf{X}^\top\mathbf{Z}) + \sum_{i=1}^{p}\mathrm{tr}(\mathbf{Z}\boldsymbol{\sigma}_{\mathbf{W}_i}\mathbf{Z}^\top) + \mathbb{E}[\mathbf{W}^\top]\mathbf{Z}^\top\mathbf{Z}\mathbb{E}[\mathbf{W}]
\end{aligned}$$

$$\begin{aligned}
&\mathrm{tr}[\mathbb{E}_{\neg\mathbf{W}}[(\mathbf{X} - \mathbf{ZW})^\top(\mathbf{X} - \mathbf{ZW})]] = \mathrm{tr}[\mathbb{E}_{\neg\mathbf{W}}(\mathbf{X}^\top\mathbf{X} - \mathbf{X}^\top\mathbf{ZW} - \mathbf{W}^\top\mathbf{Z}^\top\mathbf{X} + \mathbf{W}^\top\mathbf{Z}^\top\mathbf{ZW})] \\
&= \mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbf{X}^\top\mathbb{E}[\mathbf{Z}]\mathbf{W}) + \mathrm{tr}\left(\mathbb{E}\left[\mathbf{Z}^T\mathbf{Z}\right]\mathbf{W}\mathbf{W}^\top\right)
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[\mathrm{tr}((\mathbf{X} - \mathbf{ZW})^\top(\mathbf{X} - \mathbf{ZW}))] &= \mathbb{E}[\mathrm{tr}(\mathbf{X}^\top\mathbf{X} - \mathbf{X}^\top\mathbf{ZW} - \mathbf{W}^\top\mathbf{Z}^\top\mathbf{X} + \mathbf{W}^\top\mathbf{Z}^\top\mathbf{ZW})] \\
&= \mathrm{tr}(\mathbf{X}^\top\mathbf{X} - \mathbf{X}^\top\mathbb{E}[\mathbf{Z}]\mathbb{E}[\mathbf{W}] - \mathbb{E}[\mathbf{W}^\top]\mathbb{E}[\mathbf{Z}^\top]\mathbf{X} + \mathbb{E}[\mathbf{W}^\top\mathbf{Z}^\top\mathbf{ZW}]) \\
&= \mathrm{tr}(\mathbf{X}^\top\mathbf{X}) - 2\mathrm{tr}(\mathbf{X}^\top\mathbb{E}[\mathbf{Z}]\mathbb{E}[\mathbf{W}]) + \mathrm{tr}(\mathbb{E}[\mathbf{Z}^\top\mathbf{Z}]\mathbb{E}[\mathbf{W}\mathbf{W}^\top])
\end{aligned}$$

$$\begin{aligned}
\overline{\mathbf{R}}_{kl} :&= \mathbf{X} - \mathbb{E}[\mathbf{Z}]\left(\sum_{k'\neq k}\mathbf{e}_{k'}\mathbb{E}[\mathbf{w}_{k'}]^\top + \sum_{l'\neq l}\mathbf{e}_k\mathbb{E}[\mathbf{w}_{kl'}]^\top\right) \\
&= \mathbf{X} - \sum_{k'\neq k}\mathbb{E}[\mathbf{Z}_{k'}]\mathbb{E}[\mathbf{w}_{k'}]^\top - \sum_{l'\neq l}\mathbb{E}[\mathbf{Z}_k]\mathbb{E}[\mathbf{w}_{kl'}]^\top
\end{aligned}$$

Derivation of model parameters

In this section, we present the detailed derivation of the optimal variational distributions of variables $\mathbf{Z}, \mathbf{w}_{kl}$, and $\boldsymbol{\gamma}_{kl}$. First, we derived the log $Q(\mathbf{Z})$:

$$
\begin{aligned}
\log Q(\mathrm{Z}) = {} & \mathbb{E}_{\neg \mathrm{Z}}\left[m_c\left(\sigma^2, \sigma_0^2, \boldsymbol{\pi} \,\middle|\, \mathrm{X}, \mathrm{Z}, \mathrm{W}\right)\right] \\
& = \mathbb{E}_{\neg \mathrm{Z}}\left[\log \mathcal{MN}_{n,p}\left(\mathrm{X} \,\middle|\, \mathrm{ZW}, \mathrm{I}_n, \mathrm{I}_p\sigma^2\right)\right] + \log \mathcal{MN}_{n,k}(\mathrm{Z} \mid 0, \mathrm{I}_n, \mathrm{I}_k) \\
= {} & -\frac{\tau}{2}[-\operatorname{tr}(\mathrm{X}^\top \mathrm{Z}\mathbb{E}[\mathrm{W}]) - \operatorname{tr}(\mathbb{E}[\mathrm{W}^\top]\mathrm{Z}^\top \mathrm{X}) + \operatorname{tr}(\mathrm{Z}^\top \mathrm{Z}\mathbb{E}(\mathrm{WW}^\top))] - \frac{1}{2}\operatorname{tr}(\mathrm{Z}^\top \mathrm{Z}) + O(1) \\
= {} & -\frac{1}{2}[\operatorname{tr}(\tau \mathrm{Z}^\top \mathrm{Z}\mathbb{E}(\mathrm{WW}^\top)) + \operatorname{tr}(\mathrm{Z}^\top \mathrm{Z}) - \operatorname{tr}(\tau \mathrm{X}^\top \mathrm{Z}\mathbb{E}[\mathrm{W}]) - \operatorname{tr}(\tau \mathbb{E}[\mathrm{W}^\top]\mathrm{Z}^\top \mathrm{X})] + O(1) \\
= {} & -\frac{1}{2}[\operatorname{tr}(\mathrm{Z}^\top \mathrm{Z}(\mathbb{E}(\mathrm{WW}^\top)\tau + \mathrm{I}_k)) - \operatorname{tr}(\tau \mathrm{X}^\top \mathrm{Z}\mathbb{E}[\mathrm{W}]) - \operatorname{tr}(\tau \mathbb{E}[\mathrm{W}^\top]\mathrm{Z}^\top \mathrm{X})] + O(1) \\
= {} & -\frac{1}{2}\left[\operatorname{tr}\left(\mathrm{Z}\underbrace{(\mathbb{E}(\mathrm{WW}^\top)\tau + \mathrm{I}_k)}_{\boldsymbol{\Sigma}_\mathrm{Z}^{-1}}\mathrm{Z}^\top\right) - \operatorname{tr}(\mathrm{Z}\mathbb{E}[\mathrm{W}]\mathrm{X}^\top \tau) - \operatorname{tr}(\tau \mathrm{X}\mathbb{E}[\mathrm{W}^\top]\mathrm{Z}^\top)\right] + O(1) \\
= {} & -\frac{1}{2}\left[\operatorname{tr}\left(\mathrm{Z}\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\mathrm{Z}^\top\right) - \operatorname{tr}\left(\mathrm{Z}\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\underbrace{\boldsymbol{\Sigma}_\mathrm{Z}\mathbb{E}[\mathrm{W}]\mathrm{X}^\top \tau}_{\boldsymbol{\mu}_\mathrm{Z}^\top}\right) - \operatorname{tr}\left(\underbrace{\tau \mathrm{X}\mathbb{E}[\mathrm{W}^\top]\boldsymbol{\Sigma}_\mathrm{Z}}_{\boldsymbol{\mu}_\mathrm{Z}}\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\mathrm{Z}^\top\right)\right] + O(1) \\
= {} & -\frac{1}{2}\left[\operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\mathrm{Z}^\top \mathrm{Z}\right) - \operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\boldsymbol{\mu}_\mathrm{Z}^\top \mathrm{Z}\right) - \operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\mathrm{Z}^T\boldsymbol{\mu}_\mathrm{Z}\right) + \operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\boldsymbol{\mu}_\mathrm{Z}^T\boldsymbol{\mu}_\mathrm{Z}\right) - \operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\boldsymbol{\mu}_\mathrm{Z}^T\boldsymbol{\mu}_\mathrm{Z}\right)\right] + O(1) \\
= {} & -\frac{1}{2}\left[\operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\left(\mathrm{Z}^\top \mathrm{Z} - \mathrm{Z}^T\boldsymbol{\mu}_\mathrm{Z} - \boldsymbol{\mu}_\mathrm{Z}^T\mathrm{Z} + \boldsymbol{\mu}_\mathrm{Z}^T\boldsymbol{\mu}_\mathrm{Z}\right)\right)\right] + O(1) \\
= {} & -\frac{1}{2}\operatorname{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}(\mathrm{Z} - \boldsymbol{\mu}_\mathrm{Z})^\top(\mathrm{Z} - \boldsymbol{\mu}_\mathrm{Z})\right) + O(1) \Rightarrow
\end{aligned}
$$

$$
Q(\mathrm{Z}) = \mathcal{MN}_{n,k}(\mathrm{Z} \mid \boldsymbol{\mu}_\mathrm{Z}, \mathrm{I}_n, \boldsymbol{\Sigma}_\mathrm{Z})
$$

Second, we derive the $\log Q(w_{kl}|\boldsymbol{\gamma}_{kli} = 1)$:

$$
\begin{aligned}
\log Q(w_{kl}|\boldsymbol{\gamma}_{kli} = 1) = {} & -\frac{\tau}{2}\mathbb{E}_{\neg w_{kl}}\left[\operatorname{tr}\left(\left(\mathrm{R}_{kl} - \mathrm{Z}_k\mathbf{w}_{kl}^\top\right)^\top\left(\mathrm{R}_{kl} - \mathrm{Z}_k\mathbf{w}_{kl}^\top\right)\right)\right] - \frac{\tau_0}{2}\mathbb{E}_{\neg w_{kl}}\left[\sum_{l=1}^{L}\sum_{k=1}^{K} w_{kl}^2\right] + O(1) \\
= {} & -\frac{\tau}{2}\mathbb{E}_{\neg w_{kl}}\left[-2\operatorname{tr}\left(\mathrm{R}_{kl}^\top \mathrm{Z}_k\mathbf{w}_{kl}^\top\right) + \operatorname{tr}\left(\mathrm{Z}_k^\top \mathrm{Z}_k\mathbf{w}_{kl}^\top \mathbf{w}_{kl}\right)\right] - \frac{\tau_0}{2}w_{kl}^2 + O(1) \\
= {} & -\frac{\tau}{2}\mathbb{E}_{\neg w_{kl}}\left[-2\operatorname{tr}\left(\left(\mathrm{X} - \sum_{k' \neq k}\mathrm{Z}_{k'}\mathbf{w}_{k'}^\top - \sum_{l' \neq l}\mathrm{Z}_k\mathbf{w}_{kl'}^\top\right)^\top \mathrm{Z}_k\mathbf{w}_{kl}^\top\right) + \operatorname{tr}\left(\mathrm{Z}_k^\top \mathrm{Z}_k w_{kl}^2\right)\right] \\
& -\frac{\tau_0}{2}w_{kl}^2 + O(1) \\
= {} & -\frac{\tau}{2}\mathbb{E}_{\neg w_{kl}}\left[-2\operatorname{tr}\left(\left(\mathrm{X}^\top \mathrm{Z}_k - \sum_{k' \neq k}\mathbf{w}_{k'}\mathrm{Z}_{k'}^\top \mathrm{Z}_k - \sum_{l' \neq l}\mathbf{w}_{kl'}\mathrm{Z}_k^\top \mathrm{Z}_k\right)\mathbf{w}_{kl}^\top\right) + \operatorname{tr}\left(\mathrm{Z}_k^\top \mathrm{Z}_k w_{kl}^2\right)\right] \\
& -\frac{\tau_0}{2}w_{kl}^2 + O(1) \\
= {} & -\frac{\tau}{2}\mathbb{E}_{\neg w_{kl}}\left[-2\left(\mathrm{X}_i^\top \mathrm{Z}_k - \sum_{k' \neq k}\mathbf{w}_{k',i}\mathrm{Z}_{k'}^\top \mathrm{Z}_k - \mathrm{Z}_k^\top \mathrm{Z}_k\sum_{l' \neq l}\mathbf{w}i_{kl'}\right)w_{kl} + \mathrm{Z}_k^\top \mathrm{Z}_k w_{kl}^2\right] \\
& -\frac{\tau_0}{2}w_{kl}^2 + O(1) \\
= {} & -\frac{1}{2}\left[-2\tau\left(\mathrm{X}_i^\top \mathbb{E}[\mathrm{Z}_k] - \sum_{k' \neq k}\mathbb{E}[\mathbf{w}_{k',i}]\mathbb{E}\left[\mathrm{Z}_{k'}^\top \mathrm{Z}_k\right] - \mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]\sum_{l' \neq l}\mathbb{E}[\mathbf{w}i_{kl'}]\right)w_{kl} + \tau\mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]w_{kl}^2\right] \\
& -\frac{\tau_0}{2}w_{kl}^2 + O(1) \\
= {} & -\frac{1}{2}\left[-2\tau\left(\mathrm{X}_i^\top \mathbb{E}[\mathrm{Z}_k] - \sum_{k' \neq k}\mathbb{E}[\mathbf{w}_{k',i}]\mathbb{E}\left[\mathrm{Z}_{k'}^\top \mathrm{Z}_k\right] - \mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]\sum_{l' \neq l}\mathbb{E}[\mathbf{w}i_{kl'}]\right)w_{kl} + \tau\mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]w_{kl}^2 + \tau_0 w_{kl}^2\right] + O(1) \\
= {} & -\frac{1}{2}\left[-2\tau\left(\mathrm{X}_i^\top \mathbb{E}[\mathrm{Z}_k] - \sum_{k' \neq k}\mathbb{E}[\mathbf{w}_{k',i}]\mathbb{E}\left[\mathrm{Z}_{k'}^\top \mathrm{Z}_k\right] - \mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]\sum_{l' \neq l}\mathbb{E}[\mathbf{w}i_{kl'}]\right)w_{kl} + \underbrace{\left(\tau\mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right] + \tau_0\right)}_{1\big/\sigma_{w_{kl}}^2}w_{kl}^2\right] + O(1) \\
= {} & -\frac{1}{2\sigma_{w_{kl}}^2}\left[w_{kl}^2 - \underbrace{2\tau\sigma_{w_{kl}}^2\left(\mathrm{X}_i^\top \mathbb{E}[\mathrm{Z}_k] - \sum_{k' \neq k}\mathbb{E}[\mathbf{w}_{k',i}]\mathbb{E}\left[\mathrm{Z}_{k'}^\top \mathrm{Z}_k\right] - \mathbb{E}\left[\mathrm{Z}_k^\top \mathrm{Z}_k\right]\sum_{l' \neq l}\mathbb{E}[\mathbf{w}i_{kl'}]\right)}_{\mu_{w_{kl}}}w_{kl}\right] \Rightarrow \\
& = \log \mathcal{N}\left(\mu_{\mathbf{w}_{kl}}, \sigma_{\mathbf{w}_{kl}}^2\right)
\end{aligned}
$$

Noticed that we can update $\mathbf{w}_{kl}$ for all feature at once:

$$\boldsymbol{\mu}_{\mathbf{w}_{kl}} = \tau\sigma^2_{\mathbf{w}_{kl}}\mathbb{E}\left[\mathbf{R}_{kl}^\top \mathbf{Z}_k\right], \boldsymbol{\sigma}_{\mathbf{w}_{kl}} = \sigma^2_{\mathbf{w}_{kl}}\mathbf{I}_p$$

Finally we derive the log $Q(\boldsymbol{\gamma}_{kl})$: Note that $\tau\mathbf{R}_{kl}^\top\mathbb{E}[\mathbf{Z}_k] = \mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kl}]/\sigma^2_{w_{kl}} = \boldsymbol{\mu}_{\mathbf{w}_{kl}}/\sigma^2_{w_{kl}}$.

$$\begin{aligned}
\log Q(\boldsymbol{\gamma}_{kli} = 1) &= \mathbb{E}_{\neg\boldsymbol{\gamma}_{kl}}\left[m_c\left(\sigma^2, \sigma_0^2, \boldsymbol{\pi}, \mid \mathbf{X}, \mathbf{Z}, \mathbf{W}\right)\right] + \log\text{Multi}(\boldsymbol{\gamma}_{kl} \mid \boldsymbol{\pi}) + O(1)\\
&= -\frac{\tau}{2}\mathbb{E}_{\neg\boldsymbol{\gamma}_{kl}}\text{tr}\left(\left(\mathbf{R}_{kl} - \mathbf{Z}_k\mathbf{w}_{kl}^\top\right)^\top\left(\mathbf{R}_{kl} - \mathbf{Z}_k\mathbf{w}_{kl}^\top\right)\right) + \log\text{Multi}(\boldsymbol{\gamma}_{kl} \mid \boldsymbol{\pi}) + O(1)\\
&= -\frac{\tau}{2}\left[-2\text{tr}\left(\mathbf{R}_{kl}^\top\mathbb{E}[\mathbf{Z}_k]\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]\boldsymbol{\gamma}_{kl}^\top\right) + \text{tr}\left(\mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right]\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right]\right)\right] + \log\boldsymbol{\pi}_i + O(1)\\
&= -\frac{\tau}{2}\left[-2\mathbf{R}_{kli}^\top\mathbb{E}[\mathbf{Z}_k]\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1] + \mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right]\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kl}\right]\right] + \log\boldsymbol{\pi}_i + O(1)\langle\rangle\\
&= \tau\mathbf{R}_{kli}^\top\mathbb{E}[\mathbf{Z}_k]\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1] - \frac{\tau}{2}\mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right]\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right] + \log\boldsymbol{\pi}_i + O(1)\\
&= \frac{1}{\sigma^2_{w_{kl}}}\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 - \frac{\tau}{2}\mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right]\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right] + \log\boldsymbol{\pi}_i - \frac{\tau_0}{2}\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right] + O(1)\\
&= \frac{1}{\sigma^2_{w_{kl}}}\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 - \frac{1}{2}\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right]\left(\tau\mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right] + \tau_0\right) + \log\boldsymbol{\pi}_i + O(1)\\
&= \frac{1}{\sigma^2_{w_{kl}}}\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 - \frac{1}{2\sigma^2_{w_{kl}}}\mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right] + \log\boldsymbol{\pi}_i + O(1)\\
&= -\frac{1}{2\sigma^2_{w_{kl}}}\left[-2\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 + \mathbb{E}\left[w_{kl}^2 \mid \boldsymbol{\gamma}_{kli} = 1\right]\right] + \log\boldsymbol{\pi}_i + O(1)\\
&= -\frac{1}{2\sigma^2_{w_{kl}}}\left[-2\mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 + \sigma^2_{w_{kl}} + \mathbb{E}[w_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2\right] + \log\boldsymbol{\pi}_i + O(1)\\
&= \frac{1}{2\sigma^2_{\mathbf{w}_{kl}}}\mathbb{E}[\mathbf{w}_{kl} \mid \boldsymbol{\gamma}_{kli} = 1]^2 + \log\boldsymbol{\pi}_i + O(1) \Rightarrow
\end{aligned}$$

$$\log\tilde{\boldsymbol{\alpha}}_{kli} = \log\boldsymbol{\pi}_i - \log\mathcal{N}\left(0 \,\middle|\, \mu_{\mathbf{w}_{kl}}, \sigma^2_{\mathbf{w}_{kl}}\right)$$

$$Q(\boldsymbol{\gamma}_{kl}) = \text{Multi}(1, \boldsymbol{\alpha}_{kl} = \text{softmax}(\log\tilde{\boldsymbol{\alpha}}_{kl}))$$

In summary, the optimal variational distribution of model parameters can be summarized as:

$$Q(\mathbf{Z}) : = \mathcal{MN}_{n,k}(\mathbf{Z} \mid \boldsymbol{\mu}_\mathbf{Z}, \mathbf{I}_n, \boldsymbol{\Sigma}_\mathbf{Z}) \quad \text{(Equation 13)}$$

$$Q(w_{kl}|\gamma_{kl}) : = \mathcal{N}\left(\mu_{\mathbf{w}_{kl}}, \sigma^2_{\mathbf{w}_{kl}}\right) \quad \text{(Equation 14)}$$

$$Q(\boldsymbol{\gamma}_{kl}) : = \text{Multi}(1, \boldsymbol{\alpha}_{kl}). \quad \text{(Equation 15)}$$

The corresponding update rules for variational parameters from $Q(\cdot)$ can be expressed as,

$$\boldsymbol{\mu}_\mathbf{Z} = \tau\mathbf{X}\mathbb{E}[\mathbf{W}^\top]\boldsymbol{\Sigma}_\mathbf{Z} \quad \text{(Equation 16)}$$

$$\boldsymbol{\Sigma}_\mathbf{Z} = (\mathbb{E}[\mathbf{W}\mathbf{W}^\top]\tau + \mathbf{I}_k)^{-1} \quad \text{(Equation 17)}$$

$$\boldsymbol{\mu}_{\mathbf{w}_{kl}} = \tau\sigma^2_{\mathbf{w}_{kl}}\mathbb{E}\left[\mathbf{R}_{kl}^\top\mathbf{Z}_k\right] \quad \text{(Equation 18)}$$

$$\boldsymbol{\Sigma}_{\mathbf{w}_{kl}} = \sigma^2_{\mathbf{w}_{kl}}\mathbf{I}_p \quad \text{(Equation 19)}$$

$$\sigma^2_{\mathbf{w}_{kl}} = \left(\tau\mathbb{E}\left[\mathbf{Z}_k^\top\mathbf{Z}_k\right] + \tau_{0kl}\right)^{-1} \quad \text{(Equation 20)}$$

$$\boldsymbol{\alpha}_{kl} = \text{softmax}\left(\log\boldsymbol{\pi} - \log\mathcal{N}\left(0 \,\middle|\, \mu_{\mathbf{w}_{kl}}, \sigma^2_{\mathbf{w}_{kl}}\right)\right). \quad \text{(Equation 21)}$$

Derivation of evidence lower bound (ELBO)

The ELBO provides a natural criterion for evaluating model performance during model training, and also provides a means to perform hyperparameter optimization for model variance τ and τ_0 (or equivalently precision) parameters. Given the above definitions for Q, we derive the ELBO for SuSiE PCA as

$$\begin{aligned}\mathrm{ELBO}(\mathrm{W},\mathrm{Z}) &= \mathbb{E}_Q[\log\Pr(\mathrm{X},\mathrm{Z},\mathrm{W}) - \log Q(\mathrm{Z},\mathrm{W})]\\ &= \mathbb{E}_Q[\log\Pr(\mathrm{X}|\mathrm{Z},\mathrm{W})] + \mathbb{E}_Q[\log\Pr(\mathrm{Z},\mathrm{W}) - \log Q(\mathrm{Z},\mathrm{W})]\\ &= \mathbb{E}_Q[\log\Pr(\mathrm{X}|\mathrm{Z},\mathrm{W})] + \mathbb{E}_{Q(\mathrm{Z})}[\log\Pr(\mathrm{Z}) - \log Q(\mathrm{Z})] +\\ &\sum_{l=1}^{L}\left[\mathbb{E}_{Q(\mathbf{w}_l|\boldsymbol{\Gamma}_l)}[\log\Pr(\mathbf{w}_l|\boldsymbol{\Gamma}_l) - \log Q(\mathbf{w}_l|\boldsymbol{\Gamma}_l)] + \mathbb{E}_{Q(\boldsymbol{\Gamma}_l)}[\log\Pr(\boldsymbol{\Gamma}_l) - \log Q(\boldsymbol{\Gamma}_l)]\right]\\ &= \mathbb{E}_Q[\log\Pr(\mathrm{X}|\mathrm{Z},\mathrm{W})] + \mathbb{E}_{Q(\mathrm{Z})}[\log\Pr(\mathrm{Z}) - \log Q(\mathrm{Z})]\\ &+ \mathbb{E}_{Q(\mathrm{W},\boldsymbol{\Gamma})}[\log\Pr(\mathrm{W},\boldsymbol{\Gamma}) - \log Q(\mathrm{W},\boldsymbol{\Gamma})]\end{aligned}$$

Based on the above derivation, ELBO can be decomposed into three parts. The first term is the expectation of the data with respect to all the parameters in the model:

$$\begin{aligned}\mathbb{E}_Q[\log\Pr(\mathrm{X}|\mathrm{Z},\mathrm{W},\boldsymbol{\Gamma})] &= \mathbb{E}_Q\left[-\frac{1}{2\sigma^2}\mathrm{tr}[(\mathrm{X}-\mathrm{ZW})^\top(\mathrm{X}-\mathrm{ZW})] - \frac{np}{2}\log\left(2\pi\sigma^2\right)\right]\\ &= -\frac{1}{2\sigma^2}[\mathrm{tr}(\mathrm{X}^\top\mathrm{X}) - 2\mathrm{tr}(\mathrm{X}^\top\mathbb{E}[\mathrm{Z}]\mathbb{E}[\mathrm{W}]) + \mathrm{tr}(\mathbb{E}[\mathrm{Z}^\top\mathrm{Z}]\mathbb{E}[\mathrm{WW}^\top])] - \frac{np}{2}\log\left(2\pi\sigma^2\right)\end{aligned}$$

The second term is the negative KL divergence of Z.

$$\begin{aligned}\mathbb{E}_{Q(\mathrm{Z})}[\log\Pr(\mathrm{Z}) - \log Q(\mathrm{Z})] &= \mathbb{E}\left[-\frac{1}{2}\mathrm{tr}(\mathrm{Z}^\top\mathrm{Z}) - \frac{nk}{2}\log(2\pi) + \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Sigma}_\mathrm{Z}^{-1}(\mathrm{Z}-\boldsymbol{\mu}_\mathrm{Z})^\top(\mathrm{Z}-\boldsymbol{\mu}_\mathrm{Z})\right)\right.\\ &\left. + \frac{nk}{2}\log(2\pi) + \frac{n}{2}\log(|\boldsymbol{\Sigma}_\mathrm{Z}|)\right]\\ &= -\frac{1}{2}\mathrm{tr}(\mathbb{E}[\mathrm{Z}^\top\mathrm{Z}]) + \frac{1}{2}\mathrm{tr}\left[\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\left(E[\mathrm{Z}^\top\mathrm{Z}] - \boldsymbol{\mu}_\mathrm{Z}^\top\boldsymbol{\mu}_\mathrm{Z}\right)\right] + \frac{N}{2}\log(|\boldsymbol{\Sigma}_\mathrm{Z}|)\\ &= -\frac{1}{2}\mathrm{tr}(\mathbb{E}[\mathrm{Z}^\top\mathrm{Z}]) + \frac{1}{2}\mathrm{tr}\left[\boldsymbol{\Sigma}_\mathrm{Z}^{-1}\left(N\boldsymbol{\Sigma}_\mathrm{Z} + \boldsymbol{\mu}_\mathrm{Z}^\top\boldsymbol{\mu}_\mathrm{Z} - \boldsymbol{\mu}_\mathrm{Z}^\top\boldsymbol{\mu}_\mathrm{Z}\right)\right] + \frac{N}{2}\log(|\boldsymbol{\Sigma}_\mathrm{Z}|)\\ &= -\frac{1}{2}\mathrm{tr}(\mathbb{E}[\mathrm{Z}^\top\mathrm{Z}]) + \frac{NK}{2} + \frac{N}{2}\log(|\boldsymbol{\Sigma}_\mathrm{Z}|)\end{aligned}$$

The last term contains joint negative KL divergence of W and $\boldsymbol{\Gamma}$ can be further decomposed as following:

$$\begin{aligned}\mathbb{E}_{Q(\mathrm{W},\boldsymbol{\Gamma})}[\log\Pr(\mathrm{W},\boldsymbol{\Gamma}) - \log Q(\mathrm{W},\boldsymbol{\Gamma})] &= \mathbb{E}_{Q(\mathrm{W},\boldsymbol{\Gamma})}[\log\Pr(\mathrm{W}\mid\boldsymbol{\Gamma})\Pr(\boldsymbol{\Gamma}) - \log Q(\mathrm{W}\mid\boldsymbol{\Gamma})Q(\boldsymbol{\Gamma})]\\ &= \mathbb{E}_{Q(\mathrm{W},\boldsymbol{\Gamma})}[\log\Pr(\mathrm{W}\mid\boldsymbol{\Gamma}) - \log Q(\mathrm{W}\mid\boldsymbol{\Gamma})]\\ &+ \mathbb{E}_{Q(\mathrm{W},\boldsymbol{\Gamma})}[\log\Pr(\boldsymbol{\Gamma}) - \log Q(\boldsymbol{\Gamma})]\\ &= \sum_{k=1}^{K}\sum_{l=1}^{L}\mathbb{E}_{Q(\mathbf{w}_{kl},\boldsymbol{\gamma}_{kl})}[\log\Pr(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl}) - \log Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})] +\\ &\sum_{k=1}^{K}\sum_{l=1}^{L}\mathbb{E}_{Q(\boldsymbol{\gamma}_{kl})}[\log\Pr(\boldsymbol{\gamma}_{kl}) - \log Q(\boldsymbol{\gamma}_{kl})]\\ &= \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{i=1}^{P}\boldsymbol{\alpha}_{kli}\mathbb{E}_{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})}\left[\log\Pr(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kli}=1) - \log Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kli}=1)\right] +\\ &\sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{i=1}^{P}\mathbb{E}_{\boldsymbol{\gamma}_{kl}}[\log\Pr(\boldsymbol{\gamma}_{kli}=1) - \log Q(\boldsymbol{\gamma}_{kli}=1)]\end{aligned}$$

The first expectation term of the last line of equation $\mathbb{E}_{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})}$ can be expanded as following:

$$\mathbb{E}_{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})}\left[\log\frac{\Pr(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})}{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kl})}\right] = \sum_{i=1}^{P}\mathbb{E}_{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kli}=1)}\left[\log\frac{\Pr(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kli}=1)}{Q(\mathbf{w}_{kl}|\boldsymbol{\gamma}_{kli}=1)}\right]$$

$$= \mathbb{E}\sum_{i=1}^{P}\left[-\frac{\tau_0}{2}(w_{kli})^2+\frac{1}{2\sigma^2_{w_{kl}}}\left(w_{kli}-\boldsymbol{\mu}_{w_{kli}}\right)^2\right]$$

$$= \sum_{i=1}^{P}\left[\left(-\frac{\tau_0}{2}+\frac{1}{2\sigma^2_{w_{kl}}}\right)\left[\mu^2_{w_{kli}}+\sigma^2_{w_{kl}}\right]-\frac{1}{2\sigma^2_{w_{kl}}}\mu^2_{w_{kli}}\right]+\frac{P}{2}\log\left(\sigma^2_{w_{kl}}\tau_0\right)+\frac{P}{2}\log\left(\sigma^2_{w_{kl}}\tau_0\right)$$

$$= \sum_{i=1}^{P}\left[-\frac{\tau_0}{2}\mu^2_{w_{kli}}-\frac{\tau_0}{2}\sigma^2_{w_{kli}}+\frac{1}{2}\right]+\frac{P}{2}\log\left(\sigma^2_{w_{kl}}\tau_0\right)$$

$$-\frac{p}{2}\log(2\pi/\tau_0)+\frac{p}{2}\log\left(2\pi\sigma^2_{w_{kl}}\right)\Big]$$

$$= \sum_{i=1}^{P}\left[\left(-\frac{\tau_0}{2}+\frac{1}{2\sigma^2_{w_{kl}}}\right)\mathbb{E}\left[(w_{kli})^2\right]-\frac{1}{2\sigma^2_{w_{kl}}}\mu^2_{w_{kli}}\right]-\frac{P}{2}log(2\pi/\tau_0)+\frac{P}{2}\log\left(2\pi\sigma^2_{w_{kl}}\right)$$

And the second expectation term $\mathbb{E}_{\boldsymbol{\gamma}_{kl}}$ can be decomposed as

$$\mathbb{E}_{Q(\boldsymbol{\Gamma})}[\log\Pr(\boldsymbol{\Gamma})-\log Q(\boldsymbol{\Gamma})] = \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{i=1}^{P}\mathbb{E}_{Q(\boldsymbol{\gamma}_{kli}=1)}[(\boldsymbol{\gamma}_{kli}\log\pi_i-\boldsymbol{\gamma}_{kli}\log\alpha_{kli})]$$

$$= \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{i=1}^{P}[\mathbb{E}(\boldsymbol{\gamma}_{kli})\log(\pi_i)-\mathbb{E}(\boldsymbol{\gamma}_{kli})\log(\alpha_{kli})]$$

$$= \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{i=1}^{P}[\alpha_{kli}(\log(\pi_i)-\log\alpha_{kli})]$$

With the explicit form of ELBO, we obtained the maximum likelihood estimates of model precision parameters τ, τ_{0kl} by setting the derivative of ELBO with respect to each variance parameter to be 0, which results in closed-form update equations given by,

$$\hat{\tau}_{0kl} = \frac{\sum_{i=1}^{P}\alpha_{kli}}{\sum_{i=1}^{P}\alpha_{kli}\left(\mu^2_{w_{kli}}+\sigma^2_{w_{kli}}\right)} \quad \text{(Equation 22)}$$

$$\hat{\tau} = \frac{NP}{\sum_{i,j}X^2_{ij}-2\mathrm{tr}(\mathbb{E}[W]X^\top\boldsymbol{\mu}_Z)}. \quad \text{(Equation 23)}$$

Simulations

To investigate the performance of SuSiE PCA in variable selection and model fitting, we simulated various data sets that are controlled by 4 parameters: the sample size N, number of features P, number of latent factors K, and number of single effects L in each of the factors. For simplicity, we assume L is the same across different factors. The simulated data $\mathbf{X}$ is generated according to equation ((0.1)), where $N = 1000, P = 6000$, and $\mathbf{z}_k$ and $\mathbf{w}_k$, for $k = 1, \cdots, 4$ are simulated such that each factor only contain 40 non-zero effects (0.67%) given by,

$$\mathbf{z}_k \sim \mathcal{N}(0, \mathrm{I}_N) \quad \text{(Equation 24)}$$

$$w_{1,i} \sim \mathcal{N}(0,1) \quad i = 1, \cdots, 40 \quad \text{(Equation 25)}$$

$$w_{2,i} \sim \mathcal{N}(0,1) \quad i = 41, \cdots, 80 \quad \text{(Equation 26)}$$

$$w_{3,i} \sim \mathcal{N}\left(0, 2^2\right) \quad i = 81, \cdots, 120 \quad \text{(Equation 27)}$$

$$w_{4,i} \sim \mathcal{N}(0,1) \quad i = 121, \cdots, 160 \quad \text{(Equation 28)}$$

with the remaining effects set to zero. Considering the scale of the estimates of loadings may differ from various types of methods, we normalized the loading matrix with respect to Frobenius norm, i.e. $\mathrm{tr}(A^\top A) = \mathrm{tr}(B^\top B) = 1$.

To evaluate the accuracy of SuSiE PCA, we compared inferred posterior expectations with the true latent variables. However, due to the rotational invariance property in latent factor models, evaluating loading or latent factor accuracy can be challenging. To account for possible rotation, we leverage the Procrustes transformation,[17] which finds an orthogonal rotation matrix $\mathbf{P}$ to transform the estimated loading matrix to the true loading matrix space. Specifically, given an estimated loading matrix $\widehat{\mathbf{W}} : = \mathbb{E}_Q[\mathbf{W}]$ under approximate posterior distribution Q and true effect matrix $\mathbf{W}$, the "Procrustes Norm" can be obtained as following:

$$\|\mathbf{W} - \widehat{\mathbf{W}}\|_P^2 : = \min_{\{\mathbf{P}|\mathbf{P}^{-1} = \mathbf{P}^\top\}} \|\widehat{\mathbf{W}}\mathbf{P} - \mathbf{W}\|_F^2 \quad \text{(Equation 29)}$$

Here we perform the Procrustes analysis via Procrustes package,[16] from which $\mathbf{P}$ is obtained by performing a singular value decomposition on matrix $\widehat{\mathbf{W}}^\top\widehat{\mathbf{W}}$ (padding zeros on matrix $\widehat{\mathbf{W}}$ would ensure the above operation process correctly).

In addition, we employ the relative root mean squared error (RRMSE) to evaluate the reconstructed data loss as,

$$\text{RRMSE}(\widehat{\mathbf{X}}, \mathbf{X}) = \sqrt{\frac{\sum_{i,j}\left(\widehat{x}_{ij} - x_{ij}\right)^2}{\sum_{i,j}x_{ij}^2}} \quad \text{(Equation 30)}$$

Lastly, to assess generative modeling proficiency, we computed the log-likelihood under held-out data. Specifically, we first trained the model on simulated training data. Next, we computed latent space representations for the testing data under each of the trained models. Lastly, we computed log-likelihoods under normality assumptions given the latent representations and learned loadings and parameters.

For model comparison, we also evaluate the performance of sparse PCA[6] and Empirical Bayes Matrix Factorization (EBMF) (a recently described variational approach)[12] on the same simulation data sets with the same K, and compare the model performance with SuSiE PCA via criterion described above.

GTEx *Z* score dataset

To illustrate the application of SuSiE PCA in genetic research, we downloaded the Genotype-Tissue Expression (GTEx)[14] summary statistics data, composed of z-scores computed from the testing association between genetic variants and the gene expression levels across 44 different human tissues.[12] The GTEx project collected genotype data and gene expression data from 49 non-disease tissues across $n = 838$ individuals, providing an ideal resource database to study the relationship between genetic variants and gene expression levels.[14] The genetic variants that are statistically associated with gene expression levels are referred to as expression quantitative trait loci (eQTLs). To identify eQTLs, the GTEx project tested the association between each nearby genetic variant of a certain gene with its expression levels using linear regression to yield a Z score. The summary data we explored reflects the most significant eQTL (equivalently, the largest absolute Z score in each SNP and gene pair) at each of 16069 genes (row) from 44 tissues (column) curated from GTEx v8,[12,14] as those 16069 genes show indication of being expressed in 44 of all 49 human tissues. To identify tissue-specific components of regulatory genetic features and contributing genes, we applied SuSiE PCA across this Z score matrix with a latent dimension of 27 and the number of single effects of 18. The prior information on the number of latent dimensions comes from Wang et al. (2021)[12] who contribute to the Z score dataset and run the EBMF model with 27 factors. To determine the appropriate L that fits the data, we run the SuSiE PCA with L ranged from 10 to 25, and select the model when the increase in the total percentage of variance explained (PVE) is less than 5%. PVE is a measure of the amount of signals in the data captured by the latent component, the PVE of the factor $\mathbf{z}_k$ is calculated based on the following equation:

$$\text{PVE}_k = \frac{s_k}{\sum_k s_k + NP/\tau} \quad \text{(Equation 31)}$$

where $s_k = \sum_{i=1}^N \sum_{j=1}^P (\mathbb{E}[z_{ik}]\mathbb{E}[w_{kj}])^2$.

Purturb-seq dataset

We next investigated genome-scale Perturb-seq data[15] to discover the co-regulated gene sets affected by some common type of perturbations. The Perturb-seq data originated from Perturb-seq experiments performed by Replogle et al.[15] Perturb-seq is a cutting-edge technique combining CRISPR-based perturbations with single-cell RNA-sequencing readouts, enabling the investigation of co-regulated gene sets affected by various perturbations. The researchers employed three cell lines: K562 cells, hTERT-immortalized RPE1 cells, and HEK293T cells. CRISPRi technology was used to generate cell lines expressing dCas9-BFP-KRAB (KOX1-derived) for the perturbation experiments. Since we focus our analyses on the expression data from the K562 cell line, we give a brief description of the experiments performed on the K562 cell lines. Namely, the authors targeted genes expressed in K562 cells, transcription factors, Cancer Dependency Map common essential genes, and included non-targeting control sgRNAs accounting for 5% of the total library. The gene sets were defined based on a combination of bulk RNA-seq data from ENCODE and 10x Genomics 3′ single-cell RNA-seq data. Libraries were constructed with dual-sgRNA pairs targeting each gene, expressed from tandem U6 expression cassettes in a single lentiviral vector, and ranked based on empirical data and computational predictions. Subsequently, the author conducted Perturb-seq experiments on the K562 cells, with 2056 distinct knocked-out genes and one non-targeting control group over an average of 150 different single cells, and then measured the expression levels of the downstream 8563 genes from each cell.

The final dataset contains 310385 rows, each representing one perturbation in a specific cell, and the expression levels of 8563 downstream genes as the column. As an exploratory analysis, we omitted the single-cell level information and aggregated the expression levels of downstream genes with the same perturbation over all the cells, which resulted in a "psuedo-bulk" data matrix with 2057 rows and 8563 columns.

We then performed the SuSiE PCA and Sparse PCA to investigate the regulatory modules from the common perturbations. To exclude the batch effects and other non-genetic covariates, we regressed out the germ-line group and the mitochondrial percent from the original expression data and then aggregated the expression level of downstream genes with the same perturbation. Finally, the aggregated data is centered and standardized before input into SuSiE PCA.

As a comparison, we also run the sparse PCA with the same *K* in both datasets. While choosing an appropriate sparsity parameter alpha in sparse PCA is less straightforward than tuning *L* in the SuSiE PCA, as we cannot directly pull all of the non-zero genes even with a fairly large alpha (higher sparsity). To make a reasonable comparison, we run sparse PCA with a set of alpha from 1 to 20 and choose two models to compare: first, choose the model giving the highest PVE, then investigate the model having a similar level of PVE with SuSiE PCA.