A APPENDIX

A.1 Training parameters

For Slim, we use the proposed default candidate ordering from their manuscript and provide a minimum support threshold of 10 for pruning. For Asso, we test $T = \{0.1, 0.2, \cdots, 1\}$, as suggested in their example experiment setup in the codebase. We implemented BinaPs in Pytorch, and for all experiments train the networks using Adam [15], with an initial learning rate of 0.01, and an adaptive learning rate schedule lowering the base learning rate to 0.001 and 0.0001 after epoch 5 and 7, respectively, and train for overall 10 epochs. The exception is *Instacart*, with relatively few features but comparatively many samples. There, we observe a saturation of loss in the third epoch and hence stop after that. As discussed in the main text, Asso is trained for rank selection testing k up to the number of features in the data, similarly we set the capacity c of BinaPs to the number of features. For *Kosarak* and *Genomes*, we provide the methods with $k = c = 1000$. For medium sized data with less than $20k$ features, such as the synthetic data and the *DNA* data set, we use a batch size of 64, for all other data we use a batch size of 32. In general, we recommend these as default values as this setup proved robust about the wide range of experiments we carried out. It is however easy to carry out a parameter grid search evaluating reconstruction loss on a test set. To extract pattern sets from the network, we binarize the weights at a threshold of .2.

A.2 BinaPs with small n

Here we provide a derivation of the claim that already a single pattern is likely ($> 23\%$) to co-occur with other patterns when planting 100 patterns in 1000 samples with a density of .05 uniformly at random over the transactions. For a patterns p and any other pattern q with marginal frequencies $n_p = 50$ and $n_q = 50$, we are interested in the minimum joint frequency n_{pq} such that the pattern occur statistically significant. To test the hypothesis assuming indpendence between patterns, we use Fisher's exact test $\mathcal{F}$, setting the significance threshold to $\alpha = 0.01$. Searching for the smallest joint frequency n_{pq} such that $\mathcal{F} < \alpha$, we obtain $n_{pq} \geq 8$, meaning that if the patterns co-occur in at least 8 samples their relation is likely to be statistically significant. The next question is how likely an event of p co-occuring with any of the other 99 pattern is, which are planted in the data set. Hence we compute this probability P using the hypergeometric distribution now taking into account the overall number of patterns, yielding

$$P = 99 * \sum_{i=8}^{50} \frac{\binom{50}{i}\binom{950}{50-i}}{\binom{1000}{50}} \approx 0.233.$$

Hence, the chance of observing even just one pattern co-occuring significantly with another pattern is larger than 23%.

A.3 Instacart data

We obtained the instacart dataset from the official Kaggle challenge[4] and merged food items of the same type (e.g. all sugar of different brands) each into a single item. This allows us to circumvent problems induced by the extreme sparsity of the database, where many items only occur extremely infrequent, even just once, and thus do not expose any statistical significant relationships, and to be able to find actual patterns such as e.g. *Milk* and *Cookie*, which would not be possible if we would consider all combinations of e.g. brands and types of chocolate cookies. Treating each transaction separately, independent of time and customer id, we obtain a dataset of 1236 food items appearing in 2704831 transactions.

[4] https://www.kaggle.com/c/instacart-market-basket-analysis/data

A.4 1000 Genomes data

We processed the variant calls of all individuals available in phase 3 of the 1000 Genomes project[5], filtering for autosomal single nucleotide variants (SNVs) with an allel frequency of at least .01. For all protein coding genes specified for the reference genome, we define windows from the transcription start site (TSS) to 1000 base pairs downstream of the TSS. We then filter for SNPs that appear in such a window, and define features in our binary matrix M for all cases where at least one of the alleles show the rare variant ("1|0", "0|1", "1|1"). Thus, the data matrix is of size 3·#filtered variants × #individuals. For each individual i, we set the data entry $M_{ij} = 1$ if the individual shows genotype j.

[5] ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/