# APPENDIX

## A Proofs

### A.1 Validity of Importance Scores with Random Component

Following the notation by Candès et al. (2018) for Lemma 3.3, denoting $\boldsymbol{W}_{swap(S)} = \boldsymbol{W}([X, \tilde{X}]_{swap(S)}, Y)$ the full vector of feature statistics when swapping features in $S$, the *flip-sign property* can be summarized as: $W_{swap(S)} = \epsilon_S \odot W$ where $\odot$ is the element-wise vector multiplication and $\epsilon_S = \mathbb{I}_{j \notin S} - \mathbb{I}_{j \in S}$. As discussed by Candès et al. (2018), it should be highlighted that the final selection procedure controls FDR just because of this property. Now, by directly referring to the proof of Lemma 3.3 by Candès et al. (2018), we observe that it relies on the flip-sign property just as an equality in distribution. Therefore, with this exact same proof, we get that the result still holds when feature statistics $W$ satisfy the previous equality only in distribution. This allows us to construct valid feature statistics $W$ based on random components that are not limited to the randomness in the data itself. We can therefore also construct randomized $Z$ statistics, and we prove that the constraint mentioned earlier only needs to hold in distribution to end up with $W$ satisfying the flip-sign condition in distribution.

**Proposition A.1.** *Assume that the following equality holds in distribution for any subset $S \subset \{1, \dots, d\}$:*

$$Z([X, \tilde{X}]_{swap(S)}, Y) \overset{d}{=} Z([X, \tilde{X}], Y)_{swap(S)}$$

*Then we have the following equality in distribution:*

$$W_{swap(S)} \overset{d}{=} \epsilon_S \odot W \tag{1}$$

*Proof.* It suffices to show the result for $S = \{1\}$, as the general case can be decomposed as the concatenation of swaps of just one coordinate.

$$\begin{aligned}
&W([X, \tilde{X}]_{swap(S)}, Y) = \\
&\quad \begin{bmatrix} f_1(Z_1([X, \tilde{X}]_{swap(S)}, Y), \tilde{Z}_1([X, \tilde{X}]_{swap(S)}, Y)) \\ f_2(Z_2([X, \tilde{X}]_{swap(S)}, Y), \tilde{Z}_2([X, \tilde{X}]_{swap(S)}, Y)) \\ \vdots \\ f_d(Z_d([X, \tilde{X}]_{swap(S)}, Y), \tilde{Z}_d([X, \tilde{X}]_{swap(S)}, Y)) \end{bmatrix} \\
&\overset{d}{=} \begin{bmatrix} f_1(\tilde{Z}_1([X, \tilde{X}], Y), Z_1([X, \tilde{X}], Y)) \\ f_2(Z_2([X, \tilde{X}], Y), \tilde{Z}_2([X, \tilde{X}], Y)) \\ \vdots \\ f_d(Z_d([X, \tilde{X}], Y), \tilde{Z}_d([X, \tilde{X}], Y)) \end{bmatrix} \\
&\overset{d}{=} \begin{bmatrix} -f_1(Z_1([X, \tilde{X}], Y), \tilde{Z}_1([X, \tilde{X}], Y)) \\ f_2(Z_2([X, \tilde{X}], Y), \tilde{Z}_2([X, \tilde{X}], Y)) \\ \vdots \\ f_d(Z_d([X, \tilde{X}], Y), \tilde{Z}_d([X, \tilde{X}], Y)) \end{bmatrix} \\
&\overset{d}{=} \epsilon_S \odot W([X, \tilde{X}], Y)
\end{aligned}$$

□

### A.2 Proof of Proposition 3.1: GMM Knockoff Sampling Procedure

We prove Proposition 3.1, although this exact same proof applies in the more general setting of the Algorithm 2 in next section.

*Proof.* We consider the marginal distribution over $(X, \tilde{X})$ by summing the full joint distribution over all possible values of $K$. We then decompose the joint distribution along the sampling steps.

$$\begin{aligned}
P(X, \tilde{X}) &= \sum_{k=1}^{l} P(X, \tilde{X}, K = k) \\
&= \sum_{k=1}^{l} Q^{\tilde{X}|X,K}(\tilde{X}|X, K = k) P(X, K = k) \\
&= \sum_{k=1}^{l} Q^{\tilde{X}|X,K}(\tilde{X}|X, K = k) P^{X|K}(X|K = k) P(K = k) \\
&= \sum_{k=1}^{l} Q^{X,\tilde{X}|K}(X, \tilde{X}|K = k) P(K = k)
\end{aligned}$$

This proves exchangeability as the last line satisfies exchangeability in $(X, \tilde{X})$. □

### A.3 Comparison of Algorithm 1 and SCIP

The main contribution of our Bayesian network knockoff sampling method is due to the intractability of SCIP for a general feature distribution $P^X$ as mentioned in 3.

Indeed, SCIP sequentially samples for $1 \leq i \leq d$ the knockoff $\tilde{X}_i$ of the $i$th feature from the conditional distribution of $X_i$ given $X_{-i}, (\tilde{X}_j)_{j<i}$. That means that, at each step of the sampling process, for each sample, one needs to compute the joint distribution of $X, (\tilde{X}_j)_{j<i}$ and the conditional distribution of $X_i$, which is not computationally feasible if we assume a complex model for $P^X$.

Another difference is shown by the following: suppose that we observe variable $X$ for which we want

to sample a knockoff, and that its distribution is conditioned on a latent variable $H$. If we assume that we can construct easily the conjugates $Q^{\tilde{X}|X,H}(\tilde{x}|x,h)$ and $Q^{\tilde{H}|H,X}(\tilde{h}|h,x)$, then Algorithm 1 simplifies into Algorithm 2.

**Algorithm 2:** Knockoff Sampling Procedure for a simple Latent Variable Model

1 Sample $H \sim P^{H|X}(\cdot|X)$;
2 Sample $\tilde{H} \sim Q^{\tilde{H}|H,X}(\cdot|H,X)$;
3 Sample $\tilde{X} \sim Q^{\tilde{X}|X,H}(\cdot|X,\tilde{H})$;

We show in this simple setting that $(\tilde{H},\tilde{X})$ is not a knockoff of $(H,X)$. We write the joint distribution and decompose it along the sampling steps.

$$\begin{aligned}
&P(H,X,\tilde{H},\tilde{X}) \\
&\quad = Q^{\tilde{X}|X,H}(\tilde{X}|X,\tilde{H})Q^{\tilde{H}|H,X}(\tilde{H}|H,X) \\
&\qquad\qquad P^{H|X}(H|X)P^{X}(X) \\
&\quad = Q^{\tilde{X}|X,H}(\tilde{X}|X,\tilde{H})Q^{\tilde{H}|H,X}(H|\tilde{H},X) \\
&\qquad\qquad P^{H|X}(\tilde{H}|X)P^{X}(X)
\end{aligned}$$

To prove exchangeability of $(X,\tilde{X})$, we have to marginalize over the hidden states.

$$\begin{aligned}
&P(X,\tilde{X}) = \sum_{H,\tilde{H}} P(X,\tilde{X},H,\tilde{H}) \\
&= \sum_{H,\tilde{H}} Q^{\tilde{X}|X,H}(\tilde{X}|X,\tilde{H})Q^{\tilde{H}|H,X}(H|\tilde{H},X) \\
&\qquad\qquad P^{H|X}(\tilde{H}|X)P^{X}(X) \\
&= \sum_{\tilde{H}} Q^{\tilde{X}|X,H}(\tilde{X}|X,\tilde{H})P^{H|X}(\tilde{H}|X)P^{X}(X) \\
&= \sum_{\tilde{H}} Q^{\tilde{X}|X,H}(\tilde{X}|X,\tilde{H})P^{X|H}(X|\tilde{H})P^{H}(\tilde{H})
\end{aligned}$$

Only if we marginalize out the hidden states we get to an expression where exchangeability is satisfied for $(X,\tilde{X})$. Otherwise we don't, and therefore $(\tilde{H},\tilde{X})$ is not a knockoff of $(H,X)$. In SCIP, all the random variables sampled are part of the final knockoff sample.

Our procedure also differs from SCIP insofar it is "modular" in each local conjugate conditional. The choice of each conjugate conditional is not unique, and poor choices yield local knockoffs that are too "close" to the initial sample and decrease the power of the procedure. The worst option, which is using the feature as its own knockoff (i.e. $Q^{\tilde{i}|i,MB(i)}(\tilde{x}_i|x_{MB(i)},x_i) = \delta_{x_i=\tilde{x}_i}$) still gives valid knockoffs, though discards any possibility for that given feature to be selected. But this is why this procedure is flexible: in cases where a conditional $P^{i|MB(i)}$ has no closed form expression because of complex dependencies, we can locally choose poor conjugates and continue the procedure so that we still obtain valid knockoff samples, which is not possible when running SCIP directly as one has to sample from a complex conditional distribution that is predetermined.

We can analyze how the previous examples make use of this freedom in the choice of the conditional conjugate. In the Gaussian mixture case we could choose $Q^{\tilde{X}|X,H}(\tilde{X}|X,H) = \delta_{X=\tilde{X}}$, in which case our knockoff for $X$ would be $X$ itself, and the knockoff procedure would be powerless. In the HMM setting, after sampling the hidden nodes from the posterior given the observed ones, we could choose to keep those same nodes as local knockoffs instead of sampling a different local knockoff. In this case, the final knockoff $\tilde{X}$ we obtain is different from $X$. However, one can expect this choice to produce knockoffs with lower power, as the knockoff samples will stay "closer" to the true samples. The intuition is that if we leverage the knowledge we have about the generative process, we can sample more powerful knockoffs.

For the Hidden Markov Model, sample:

- $H \sim P^{H|X}(\cdot|X)$, we sample the hidden states. Conditionally on $X$ the distribution of $H$ is that of a Markov chain.
- $\tilde{H} \sim Q^{\tilde{H}|H,X}(\cdot|H,X)$ we sample a new knockoff Markov chain via SCIP.
- $\tilde{X} \sim Q^{\tilde{X}|X,H}(\cdot|X,\tilde{H})$. However, $Q^{\tilde{X}|X,H}(\tilde{x}|x,h)$ is constructed based on the distribution $P^{X|H}(x|h)$. But because of the structure of the HMM, the observed states are independent conditionally on the hidden states. If the observed states are univariate then we can simplify the conjugate conditional and sample $\tilde{X} \sim Q^{\tilde{X}|X,H}(\cdot|X,\tilde{H}) = P^{X|H}(\cdot|\tilde{H})$ (see comments after Definition 3.1).

**Example: Sampling Knockoffs for LDA** As an additional example, we describe how Algorithm 1 works to generate knockoffs from Latent Dirichlet Allocation (LDA). We use the same notation for LDA as defined by Blei et al. (2003): the $n$th word $W_{dn}$ in document $d$ is sampled from a multinomial distribution parametrized by $\beta_{Z_{dn}}$, where $Z_{dn}$ corresponds to the topic assignment for $W_{dn}$. Topic assignment is sampled from a multinomial distribution parametrized by $\theta_d$, the distribution of topics in document $d$. Finally $\theta_d$ for each document is sampled from a Dirichlet distribution with hyperparameter $\alpha$.

1. The first step to build knockoffs is to learn the parameters $\alpha, \beta$ of the model, which can be done by variational EM (Blei et al., 2003). Then, we need to sample the hidden variables $Z_{dn}, \theta_d$ given the observed ones $W_{dn}$ : this is an inference problem for which direct computation is intractable, but we can approximate that posterior distribution via standard variational Bayes methods.
2. Next is to sample local knockoffs. This is exactly analog to one pass of Gibbs sampling over the whole DAG following a topological ordering, except that instead of sampling with respect to the conditional distribution of the node given its Markov blanket, we sample from the conjugate conditional distribution, conditioning on the appropriate variables as explained in Algorithm (1). We sample each $\theta_d$ based on the conjugate conditional distribution. However, as $\theta_d$ is Dirichlet, any given coordinate is determined by the others, so the only possible choice is to set $\tilde{\theta}_d = \theta_d$. Then, as $Z_{dn}$ is univariate, its conjugate conditional simplifies too so that we just sample from the local conditional probability, and so on for $W_{dn}$.

### A.4 Proof of Theorem 3.2: DAG Knockoff Sampling Procedure

*Proof.* The joint probability distribution can be decomposed as follows by following the sampling steps:

$$P(X, \tilde{X}) = P(X) \prod_{i=1}^{m} Q^{\tilde{i}|i,MB(i)}(\tilde{X}_i | \tilde{X}_{\{1:i-1\} \cap MB(i)}, X_{\{i+1:m\} \cap MB(i)}, X_i) \quad (2)$$

In order to show that $(X_O, \tilde{X}_O)$ is exchangeable, we want to show that if we marginalize out this joint distribution with respect to the hidden states $(X_H, \tilde{X}_H)$, we get an exchangeable distribution.

We first show that, iterating recursively over all the nodes, and summing over all values of $X_H$, we obtain the following expression.

$$\sum_{X_H} P(X, \tilde{X}) = P(\tilde{X}) \prod_{i \in O} Q^{\tilde{i}|i,MB(i)}(X_i | \tilde{X}_{\{1:i-1\} \cap MB(i)}, \tilde{X}_i) \quad (3)$$

For simplicity, here we consider discrete random variables, so that marginalizing the joint distribution over $X_i$ means summing over all possible values of $X_i$. Everything stays valid for continuous random variables, replacing sums by integrals. Starting from equation (2), which corresponds to step 1, we do sequentially $m$ steps to get to (3). Suppose that at step $1 \leq k \leq m$ we have the following equality where the left-hand term is the product of the right-hand terms:

$$\begin{aligned} &\sum_{\substack{X_l \\ l \in H,\, l \leq k-1}} P(X, \tilde{X}) = P(\tilde{X}_{1:k-1}, X_{k:m}) \\ &\times \prod_{i \geq k} Q^{\tilde{i}|i,MB(i)}(\tilde{X}_i | \tilde{X}_{\{1:i-1\} \cap MB(i)}, X_{\{i+1:m\} \cap MB(i)}, X_i) \\ &\times \prod_{\substack{i \in O \\ i \leq k-1}} Q^{\tilde{i}|i,MB(i)}(X_i | \tilde{X}_{\{1:i-1\} \cap MB(i)}, \tilde{X}_i) \end{aligned}$$

The key element is that, by following the topological order, at step $k$ the variable $X_k$ only appears in the joint probability $P(\tilde{X}_{1:k-1}, X_{k:m})$ and in the term

$$Q^{\tilde{k}|k,MB(k)}(\tilde{X}_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}, X_k)$$

(Notice that, if $i \leq k-1$ corresponds to an observed node, it has no descendents. Therefore the Markov blanket of such node is a subset of the nodes with smaller index/topological ordering). We isolate these two terms and start by writing down the joint probability as a conditional probability. By definition of the Markov blanket, we can simplify the expression of the conditional probability. Then, we obtain two terms that are conjugate in the exchangeable sense.

$$\begin{aligned} &P(\tilde{X}_{1:k-1}, X_{k:m}) Q^{\tilde{k}|k,MB(k)}(\tilde{X}_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}, X_k) \\ &= P^{k|MB(k)}(X_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}) \\ &\times P(\tilde{X}_{\{1:k-1\}}, X_{\{k+1:m\}}) \\ &\times Q^{\tilde{k}|k,MB(k)}(\tilde{X}_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}, X_k) \\ &= P^{k|MB(k)}(\tilde{X}_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}) \\ &\times P(\tilde{X}_{\{1:k-1\}}, X_{\{k+1:m\}}) \\ &\times Q^{\tilde{k}|k,MB(k)}(X_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}, \tilde{X}_k) \\ &= P(\tilde{X}_{1:k}, X_{k+1:m}) Q^{\tilde{k}|k,MB(k)}(X_k | \tilde{X}_{\{1:k-1\} \cap MB(k)}, X_{\{k+1:m\} \cap MB(k)}, \tilde{X}_k) \end{aligned}$$

We swap in the previous expression the two terms and we get the following product:

$$\sum_{\substack{X_l \\ l\in H,\, l\leq k-1}} P(X,\tilde{X}) = P(\tilde{X}_{1:k}, X_{k+1:m})$$
$$\times \prod_{i\geq k+1} Q^{\tilde{i}|i,MB(i)}(\tilde{X}_i|\tilde{X}_{\{1:i-1\}\cap MB(i)}, X_{\{i+1:m\}\cap MB(i)}, X_i)$$
$$\times \prod_{\substack{i\in O \\ i\leq k-1}} Q^{\tilde{i}|i,MB(i)}(X_i|\tilde{X}_{\{1:i-1\}\cap MB(i)}, \tilde{X}_i)$$
$$\times Q^{\tilde{k}|k,MB(k)}(X_k|\tilde{X}_{\{1:k-1\}\cap MB(k)}, X_{\{k+1:m\}\cap MB(k)}, \tilde{X}_k)$$

If $k \in H$, then we sum both sides of the equality over $X_k$. But now, $X_k$ only appears in the last term, and summing over it gives 1. If we reach a node with no descendents, i.e. $k \in O$, then we do not marginalize out. However we have the following simplification:

$$Q^{\tilde{k}|k,MB(k)}(X_k|\tilde{X}_{\{1:k-1\}\cap MB(k)}, X_{\{k+1:m\}\cap MB(k)}, \tilde{X}_k) = Q^{\tilde{k}|k,MB(k)}(X_k|\tilde{X}_{\{1:k-1\}\cap MB(k)}, \tilde{X}_k)$$

In both cases, we get to the next step in our recursion. After completing last step, we get to equation (3). Notice that this expression is exchangeable in $X_O, \tilde{X}_O$, for every assignment of $\tilde{X}_H$. Indeed, for $l \in O$, we have that $\{1 : l-1\} \cap MB(l) = MB(l)$, so

$$\sum_{X_H} P(X,\tilde{X}) = P(\tilde{X}_{-l}) P^{l|MB(l)}(\tilde{X}_l|\tilde{X}_{MB(l)})$$
$$\times Q^{\tilde{l}|l,MB(l)}(X_l|\tilde{X}_{\{1:l-1\}\cap MB(l)}, \tilde{X}_l)$$
$$\times \prod_{\substack{i\in O \\ i\neq l}} Q^{\tilde{i}|i,MB(i)}(X_i|\tilde{X}_{\{1:i-1\}\cap MB(i)}, \tilde{X}_i)$$

And $(X_l, \tilde{X}_l)$ do not appear in the last product term as two observed nodes cannot be in the Markov blanket of each other. $(X_l, \tilde{X}_l)$ only appear in the conjugate probabilities, therefore the exchangeability in $(X_l, \tilde{X}_l)$ holds. Again, as two observed nodes cannot appear in the Markov blanket of the other, this step can be repeated for different indices $l \in O$, hence the exchangeability of the expression. This symmetry is at fixed values of $\tilde{X}_H$. Therefore, it still holds when we sum over $\tilde{X}_H$. Hence

$$P(X_O, \tilde{X}_O) = \sum_{\tilde{X}_H, X_H} P(X,\tilde{X})$$

satisfies exchangeability. □

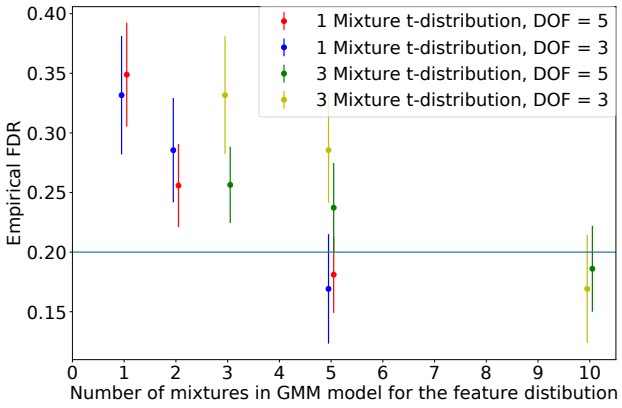


Figure 4: **Fitting a t-distribution with a Mixture of Gaussians** We evaluate the empirical FDR when running the knockoff procedure with a misspecified model. We generate knockoffs by fitting a Gaussian mixture in settings where the features are from a mixture of t-distribution, for different degrees of freedom (DOF).

## B Robustness of the Procedure to Model Misspecification

As explained when first introducing the *Model-X* knockoff procedure in Candès et al. (2018), instead of considering a model for the conditional distribution of $Y|X$, all the assumptions are related to modeling $X$. The *burden of knowledge* shifts from $Y|X$ to $X$. The same way valid p-values rely on assumptions on $Y|X$, (parametric model, noise distribution, asymptotic regime...), valid knockoffs rely on assumptions on the distribution of $X$: mainly, that we can approximate it very well.

When we generate knockoffs based on a Gaussian mixture model, and more generally a Bayesian network, we assume that these probabilistic models are good approximations for $P_X$, and that they can be properly fitted. This is a very strong assumption, as not only the model we use to represent $X$ may be incorrect, but the estimated parameters of the model depend on the fitting procedure, which sometimes provides only an approximation to the actual distribution encoded by the Bayesian network (as when using Variational Inference). Optimization methods commonly used such as Expectation-Maximization (EM) can also get stuck in local minima. However, the knockoff procedure is remarkably robust when dealing with these issues. Existing theoretical robustness bounds (Barber et al., 2018) are based on controlling the KL-divergence of the model with respect to the true $P_X$. This can help explain why EM in our method, and hopefully other fitting procedures, yield knockoffs that are somehow

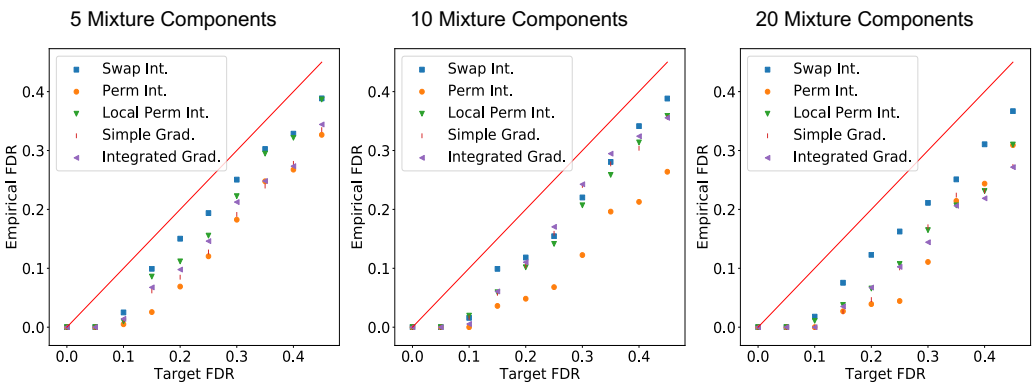


Figure 5: **Empirical FDR for Experiments With Synthetic Features**

valid: these methods minimize the KL-divergence of the model with respect to the distribution of $X$.

## C Synthetic Data Generation and FDR Control

As an example, we provide simulations showing the empirical FDR for a mixture of t-distributions in Figure 4 (at a fixed target FDR 0.2), as a function of the number of Gaussian mixtures we use to model the distribution. The conclusion is that, with enough mixtures, the knockoff procedure is able to control FDR, even though we cannot expect to correctly represent any t-distribution through a mixture of Gaussians.

To generate a synthetic data set with $n$ samples, $d$ features, $l$ mixtures and $C$ different classes, we implemented the following steps:

- We generate random values for the means and covariance matrices (using scikit-learn positive-semidefinite matrix generation function) for each of the $l$ mixtures and the mixture proportions.
- For each $1 \leq i \leq n$ we sample $K_i$ the mixture assignment for sample $i$.
- We sample $X = (X_{i1}, \ldots, X_{id})$ from the Gaussian distribution corresponding to the $K_i$th mixture.
- We define $f_c : \mathbb{R}^d \to \mathbb{R}$ for $c \in \{1, \ldots, C\}$ to be 3rd order polynomial functions over the attributes. The coefficients of the polynomial functions are randomly sampled from $\mathcal{N}(0, 1)$.
- Each sample $X_i$ is labeled by $Y_i = \arg\max_{c \in \{1,\ldots,C\}}(f_c(X_i) + \epsilon_{ic})$ where $\epsilon_{ic} \sim \mathcal{N}(0, 0.1)$ i.i.d.

To generate synthetic labels for a real world data set, we go through the same procedure, but without generating the input features. It is crucial to notice that, in these experiments with real data, it is not possible to verify that our method controls FDR, given that we can not obtain new batches of data coming from the same distribution on which to repeat the procedure to get an empirical FDR. These experiments are done for the purpose of power comparison, which remains pertinent even if we only regenerate the synthetic label.

For the experiments where we repeatedly generated the synthetic data $X$, we can verify that our procedure controls FDR by computing an empirical FDR over several runs of the procedure. Figure 5 plots the empirical FDR vs. the target FDR, whenever $X$ is sampled from different numbers of mixtures, and for all the methods we use to compute feature statistics.

## D Selected Features for Real Datasets

We work on three real world data sets: (1) 17596 randomly sampled participants from the UK Biobank data set (Sudlow et al., 2015). Each individual has 284 phenotype features. (2) Bank Marketing (Moro et al., 2014) Data Set of UCI (Dheeru and Karra Taniskidou, 2017), containing 45211 samples with 10 real-valued features for a binary classification task of bank telemarketing success prediction. (3) Polish bankruptcy dataset (Zięba et al., 2016) of the UCI repository containing 10503 samples with no missing attribute, each with 64 real-valued attributes for a binary task of company bankruptcy prediction.

**Disease Prediction** With target FDR=0.3, the following features were selected for the task of Malignant neoplasm of breast with ICD10 code C50:

- Duration of walks
- Ankle spacing width
- Average weekly champagne plus white wine intake
- Coffee intake
- Number of cigarettes previously smoked daily
- Interval between previous point and current one in numeric path (trail # 1) (related to intelligence

question results)
- Father's age at death
- Longest period of depression
- Particulate matter air pollution
- Inverse distance to the nearest road
- Number of days/week walked +10 minutes
- Mean reticulocyte volume
- Length of menstrual cycle
- Average weekly spirits intake

**Bank Marketing Success Prediction** With target FDR=0.3:

- age
- duration
- campaign
- pdays
- previous
- emp.var.rate
- cons.price.idx
- cons.conf.idx
- euribor3m'
- nr.employed

**Bankruptcy Prediction** With target FDR=0.3:

- Gross profit (in 3 years) / total assets
- Profit on sales / total assets
- Retained earnings / total assets
- Gross profit / short-term liabilities

## E Intuition Behind Drawback of Permutation Importance Scores

We explain the phenomenon through Figure 6.

(a) A neural network is trained with a dataset with one feature concatenated with its generated knockoff feature. The horizontal axis corresponds to the original and the vertical axis is the knockoff feature. The decision boundaries of the trained network are displayed.

(b) Applying shuffling to one of the samples in its original feature will result in an incorrect prediction and therefore a high importance score for the original feature.

(c) Although the knockoff feature has no effect on prediction, as applying shuffling results in an *off-distribution* fake data point, the predicted label of the fake data point will be incorrect again as it lies in part of the input space that the network has not been trained on. The importance score for both the original and the knockoff feature will both be high, which will result in a small feature statistic and therefore prevent that non-null feature from being selected.

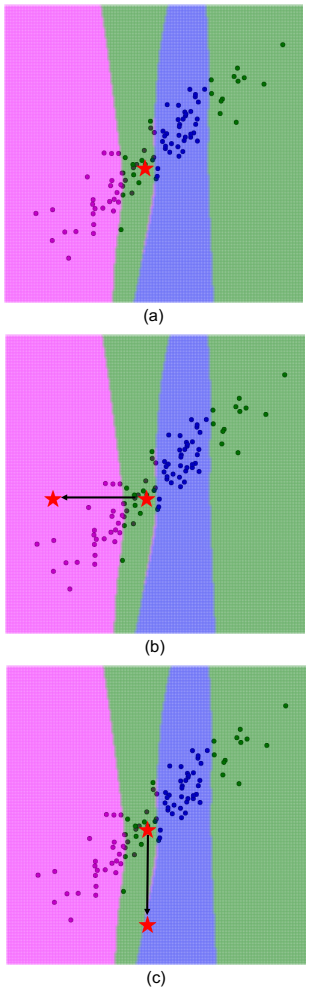


Figure 6: **Drawback of Permutation Method for Importance Score**