Mathematical Formulations for Synthetic Data Generation with Controlled Leakage

1. Feature Generation ($\boldsymbol{X}$)

We generate the feature matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ by sampling n observations from a multivariate normal distribution:

$$\boldsymbol{x}_i \sim \mathcal{N}\left(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x\right), \quad i = 1, \ldots, n$$

- $\boldsymbol{\mu}_x \in \mathbb{R}^d$: Mean vector of the features.
- $\boldsymbol{\Sigma}_x \in \mathbb{R}^{d \times d}$: Covariance matrix of the features.

2. Ground Truth Concepts ($\boldsymbol{c}_i$)

We construct the ground truth concepts $\boldsymbol{c}_i \in \{0, 1\}^k$ as follows:

2.1. Constructing Matrix A

Matrix $\mathbf{A} \in \mathbb{R}^{k \times d}$ is designed to project the first b features into the concept space:

$$\mathbf{A} = \left[\mathbf{R}_A \mid \mathbf{0}_{k \times (d-b)}\right]$$

- $\mathbf{R}_A \in \mathbb{R}^{k \times b}$: Random projection matrix with entries:

$$\left(\mathbf{R}_A\right)_{jp} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1), \quad j = 1, \ldots, k; \quad p = 1, \ldots, b$$

- $\mathbf{0}_{k \times (d-b)}$: Zero matrix to exclude the remaining features.

2.2. Computing Success Probabilities $\boldsymbol{\pi}_i$

We compute the logits for the ground truth concepts:

$$\boldsymbol{\eta}_i = \mathbf{A}\boldsymbol{x}_i + \boldsymbol{\epsilon}_c$$

- $\boldsymbol{\epsilon}_c \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_c\right)$: Noise vector with covariance $\boldsymbol{\Sigma}_c \in \mathbb{R}^{k \times k}$.

  The success probabilities are then obtained via the sigmoid function:

$$\boldsymbol{\pi}_i = \sigma\left(\boldsymbol{\eta}_i\right)$$

- Sigmoid function $\sigma(z)$:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

2.3. Sampling Ground Truth Concepts $\boldsymbol{c}_i$

Each concept c_{ij} is sampled from a Bernoulli distribution:

$$c_{ij} \sim \text{Bernoulli}\left(\pi_{ij}\right), \quad j = 1, \ldots, k$$

3. Leakage Term ($\boldsymbol{l}_i$)

We construct the leakage term $\boldsymbol{l}_i \in \mathbb{R}^k$ as follows:

3.1. Constructing Matrix B

Matrix $\mathbf{B} \in \mathbb{R}^{k \times d}$ projects selected features into the concept space:

$$\mathbf{B} = [\mathbf{0}_{k \times b} \mid \mathbf{R}_B \mid \mathbf{0}_{k \times l}]$$

- $\mathbf{0}_{k \times b}$: Zero matrix to exclude the first b features.
- $\mathbf{R}_B \in \mathbb{R}^{k \times (d-b-l)}$: Random projection matrix with entries:

$$(\mathbf{R}_B)_{jq} \overset{\text{iid}}{\sim} \mathcal{N}(0,1), \quad j = 1, \dots, k; \quad q = 1, \dots, d-b-l$$

- $\mathbf{0}_{k \times l}$: Zero matrix to exclude the last l features.

3.2. Computing Leakage Term $\boldsymbol{l}_i$

$$\boldsymbol{l}_i = \mathbf{B}\boldsymbol{x}_i$$

4. Estimated Concepts ($\hat{\boldsymbol{c}}_i$)

We compute the estimated concepts $\hat{\boldsymbol{c}}_i \in [0,1]^k$ as:

$$\hat{\boldsymbol{c}}_i = \sigma\left(\mathbf{A}\boldsymbol{x}_i + \boldsymbol{l}_i + \boldsymbol{\epsilon}_{\hat{c}}\right)$$

- $\boldsymbol{\epsilon}_{\hat{c}} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{\hat{c}}\right)$: Noise vector with covariance $\boldsymbol{\Sigma}_{\hat{c}} \in \mathbb{R}^{k \times k}$.
- σ: Sigmoid function as defined earlier.

5. Target Labels (y_i)

We generate the target labels $y_i \in \{1, \dots, J\}$ as follows:

5.1. Defining Function f

If no custom function f is provided, we define $f : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}^J$ using a simple Multi-Layer Perceptron (MLP):

- **Inputs**: Concatenate $\boldsymbol{c}_i$ and $\boldsymbol{l}_i$:

$$\boldsymbol{u}_i = \begin{bmatrix} \boldsymbol{c}_i \\ \boldsymbol{l}_i \end{bmatrix} \in \mathbb{R}^{2k}$$

- **First Layer**:

$$\boldsymbol{h}_i = \phi\left(\mathbf{W}_1\boldsymbol{u}_i + \boldsymbol{b}_1\right) \in \mathbb{R}^h$$

  - $\mathbf{W}_1 \in \mathbb{R}^{h \times 2k}$: Weight matrix with entries $\sim \mathcal{N}(0,1)$.
  - $\boldsymbol{b}_1 \in \mathbb{R}^h$: Bias vector initialized to zeros.
  - $\phi(z)$: ReLU activation function:

$$\phi(z) = \max(0, z)$$

- **Output Layer**:

$$\boldsymbol{o}_i = \mathbf{W}_2\boldsymbol{h}_i + \boldsymbol{b}_2 \in \mathbb{R}^J$$

  - $\mathbf{W}_2 \in \mathbb{R}^{J \times h}$: Weight matrix with entries $\sim \mathcal{N}(0,1)$.
  - $\boldsymbol{b}_2 \in \mathbb{R}^J$: Bias vector initialized to zeros.

5.2. Computing Target Probabilities $\boldsymbol{p}_i$

We compute the logits for the target probabilities:

$$\boldsymbol{z}_i = f\left(\boldsymbol{c}_i, \boldsymbol{l}_i\right) + \boldsymbol{\epsilon}_y$$

- $\boldsymbol{\epsilon}_y \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_y\right)$: Noise vector with covariance $\boldsymbol{\Sigma}_y \in \mathbb{R}^{J \times J}$.

We then apply the softmax function to obtain the probabilities:

$$\boldsymbol{p}_i = \operatorname{softmax}\left(\boldsymbol{z}_i\right)$$

- Softmax function:

$$\operatorname{softmax}\left(\boldsymbol{z}_i\right)_j = \frac{\exp\left(z_{ij}\right)}{\sum_{k=1}^{J} \exp\left(z_{ik}\right)}, \quad j = 1, \ldots, J$$

5.3. Sampling Target Labels y_i

We sample y_i from a categorical distribution based on $\boldsymbol{p}_i$:

$$y_i \sim \operatorname{Categorical}\left(\boldsymbol{p}_i\right)$$

This results in $y_i \in \{1, 2, \ldots, J\}$.

6. Summary of Variables and Parameters

- n: Number of observations.
- d: Dimensionality of features.
- k: Number of concepts.
- J: Number of target classes.
- b: Number of features used in the ground truth concepts.
- l: Number of features excluded from leakage.
- $\boldsymbol{\mu}_x \in \mathbb{R}^d$: Mean vector of features.
- $\boldsymbol{\Sigma}_x \in \mathbb{R}^{d \times d}$: Covariance matrix of features.
- $\boldsymbol{\Sigma}_c \in \mathbb{R}^{k \times k}$: Covariance matrix of noise in ground truth concepts.
- $\boldsymbol{\Sigma}_{\hat{c}} \in \mathbb{R}^{k \times k}$: Covariance matrix of noise in estimated concepts.
- $\boldsymbol{\Sigma}_y \in \mathbb{R}^{J \times J}$: Covariance matrix of noise in target logits.
- f: Function mapping concepts and leakage to target logits.