# Implementation of g in Nonlinear Combination

## 1. Forms of Interaction Terms

In nonlinear combinations, the function $$\( g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) \)$$ is used to model the interaction between different conditions $$\( \tilde{d}\_i \) and \( \tilde{d}\_j \)$$. Here are some common implementation approaches:

### (1) Polynomial Interaction

Capture the interaction using a simple polynomial form:

$$

g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) = \tilde{d}\_i \cdot \tilde{d}\_j \quad \text{(quadratic interaction)}

$$

Or extend to higher-order polynomials:

$$

g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) = \tilde{d}\_i^p \cdot \tilde{d}\_j^q

$$

where \( p, q \) are non-negative integers that control the degree of nonlinearity.

\*\*Advantages:\*\* Simple and intuitive.

\*\*Disadvantages:\*\* Higher-order polynomials may lead to overfitting.

### (2) Kernel-Based Methods

To capture complex relationships in high-dimensional spaces, use kernel functions:

$$

g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) = \exp\left(-\frac{(\tilde{d}\_i - \tilde{d}\_j)^2}{\sigma^2}\right)

$$

- \*\*Gaussian Kernel:\*\* Measures the similarity between two conditional distances.

- $${\sigma}$$: Controls the range of influence for the interaction.

### (3) Attention Mechanism

Leverage attention to dynamically adjust the weights of interactions:

$$

g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) = \text{softmax}\_j\left(\frac{\tilde{d}\_i \cdot \tilde{d}\_j}{\sqrt{d}}\right) \cdot \tilde{d}\_j

$$

- \*\*Softmax:\*\* Ensures normalized interaction weights.

- Automatically learns dependencies between every pair of conditions.

### (4) Neural Network-Based Nonlinear Mapping

Embed condition interactions in a neural network to automatically learn complex relationships:

$$

g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) = \text{MLP}([\tilde{d}\_i, \tilde{d}\_j])

$$

- Concatenate the two condition distances as input to a multi-layer perceptron (MLP).

- The MLP captures high-order nonlinear interactions.

## 2. Combination Strategy

The complete nonlinear combination can be formulated as:

$$

E(\mathbf{d}) = \sum\_{i=1}^n w\_i \cdot f\_i(\tilde{d}\_i) + \sum\_{i \neq j} w\_{ij} \cdot g\_{ij}(\tilde{d}\_i, \tilde{d}\_j)

$$

- $$\( f\_i(\tilde{d}\_i) \)$$: A nonlinear transformation for individual conditions (e.g., $$\( \exp(-\tilde{d}\_i^2) \)$$).

- $$\( g\_{ij}(\tilde{d}\_i, \tilde{d}\_j) \)$$: Captures interactions between conditions.

---

# Condition Fusion with Covariance Matrix

## 1. Meaning of Covariance Matrix

The covariance matrix \( \Sigma \) is a symmetric matrix that captures the linear dependencies between conditions. For \( n \) conditions \( \mathbf{d} = [d\_1, d\_2, \dots, d\_n] \), the covariance matrix is defined as:

$$

\Sigma\_{ij} = \text{Cov}(d\_i, d\_j) = \mathbb{E}[(d\_i - \mu\_i)(d\_j - \mu\_j)]

$$

- $$\( \mu\_i \)$$: The mean of condition $$\( d\_i \)$$.

- If $$\( \Sigma\_{ij} > 0 \), \( d\_i \)$$ and $$\( d\_j \)$$ are positively correlated; if $$\( \Sigma\_{ij} < 0 \)$$, they are negatively correlated.

## 2. Methods to Estimate the Covariance Matrix

### (1) Sample Estimation

Given a set of samples \( \{ \mathbf{d}^{(k)} \}\_{k=1}^N \), the covariance matrix can be estimated as:

$$

\hat{\Sigma}\_{ij} = \frac{1}{N-1} \sum\_{k=1}^N \left(d\_i^{(k)} - \bar{d}\_i\right)\left(d\_j^{(k)} - \bar{d}\_j\right)

$$

- $$\( d\_i^{(k)} \)$$: The value of condition $$\( i \)$$ in the $$\( k \)$$-th sample.

- $$\( \bar{d}\_i \)$$: The sample mean of condition $$\( i \)$$.

### (2) Regularized Covariance Matrix

To prevent instability due to insufficient samples or noise, regularization can be applied:

$$

\hat{\Sigma} = (1 - \lambda) \hat{\Sigma} + \lambda I

$$

- $$\( \lambda \in [0, 1] \)$$: Regularization parameter.

-$$ \( I \)$$: Identity matrix, ensuring the covariance matrix is positive definite.

### (3) Kernel-Based Estimation

For conditions that have nonlinear dependencies, kernel techniques can be used to estimate covariance:

$$

K\_{ij} = \phi(d\_i)^\top \phi(d\_j)

$$

- $$\( \phi(d) \)$$: A mapping function that embeds the conditions into a high-dimensional space.

- $$\( K\_{ij} \)$$: Kernel matrix, approximating the covariance in the high-dimensional space.

## 3. Energy Function Using the Covariance Matrix

Once the covariance matrix is obtained, the energy function can be defined as:

$$

E(\mathbf{d}) = \mathbf{d}^\top \Sigma^{-1} \mathbf{d}

$$

- $$\( \Sigma^{-1} \)$$: The inverse covariance matrix (precision matrix), capturing the dependency structure between conditions.

\*\*Significance:\*\*

- If the conditions are independent, $$\( \Sigma \)$$ is diagonal.

- If the conditions are dependent, the off-diagonal elements of $$\( \Sigma^{-1} \)$$ reflect the strength of the dependencies.

## 4. Improvements to the Covariance Matrix

### (1) Sparse Covariance

In high-dimensional settings, enforce sparsity to retain only the most important dependencies:

$$

\min\_\Sigma \|\Sigma^{-1}\|\_1 \quad \text{s.t. Covariance Constraints}

$$

- $$\( \|\cdot\|\_1 \)$$: Sparsity regularization.

### (2) Dimensionality Reduction

Use techniques like Principal Component Analysis (PCA) to reduce the condition space before estimating the covariance matrix.

### (3) Dynamic or Time-Varying Conditions

For time-series or dynamic conditions, extend the covariance matrix to a conditional covariance tensor to model temporal dependencies.

---

# Summary

- \*\* g in Nonlinear Combination:\*\*

- Can be implemented using polynomials, kernels, attention mechanisms, or neural networks.

- Match the complexity of $$\( g\_{ij} \)$$ to the dependency degree of the conditions to avoid overfitting.

- \*\*Covariance Matrix Estimation:\*\*

- Use sample-based estimation, regularization, or kernel methods depending on the condition properties.

- The covariance matrix is suitable for capturing dependencies in low-dimensional or strongly correlated conditions.

- \*\*Practical Recommendations:\*\*

- Choose $$\( g\_{ij} \)$$'s complexity based on the application and data characteristics.

- In high-dimensional scenarios, apply sparsity or dimensionality reduction to the covariance matrix.