| **Approach** | **High-Level Definition** | **Advantages** | **Limitations** |
| --- | --- | --- | --- |

|  |  |  |  |
| --- | --- | --- | --- |
| **Random Generation through Script** | Creating synthetic data by writing scripts that generate data points randomly based on defined rules and distributions. | - Simple and easy to implement<br>- Requires minimal computational resources<br>- Can be highly customizable | - May lack realism and complexity<br>- Hard to mimic intricate patterns in real data<br>- Limited scalability |

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Masking** | Protecting sensitive data by replacing it with altered versions while maintaining the data’s utility for testing and analysis. | - Maintains the structure and format of original data<br>- Helps in compliance with data privacy regulations<br>- Simple to implement | - May not fully hide all sensitive information<br>- Limited to the structure of the original dataset |

|  |  |  |  |
| --- | --- | --- | --- |
| **Generative Adversarial Networks (GANs)** | Using neural networks to generate new data samples that resemble the training data by learning its distribution. | - Can generate highly realistic data<br>- Captures complex data distributions<br>- Scalable to large datasets | - Computationally intensive<br>- Requires a lot of data for training<br>- Can be difficult to train effectively (e.g., mode collapse) |