Hyperparameter tuning, also known as optimization, is the process of finding the best combination of hyperparameters for your machine learning model. These hyperparameters control the overall learning process and significantly impact the model's performance. Here are some different types of hyperparameter tuning methods:

1. Manual Search:
    - This involves manually trying different combinations of hyperparameter values and evaluating the model's performance on each one.
    - Suitable for small datasets and models with few hyperparameters, but can be time-consuming and inefficient for larger setups.
2. Grid Search:
    - This method systematically iterates through a predefined grid of all possible combinations of hyperparameter values.
    - Evaluates the model on each combination and identifies the one with the best performance.
    - Ensures thorough exploration but can be computationally expensive, especially with large grids.
3. Random Search:
    - This randomly samples potential hyperparameter values within a defined range.

- More efficient than grid search for large grids and can sometimes find better optima due to its wider exploration.
- However, it may miss certain regions of the hyperparameter space and require more evaluations.

4. Bayesian Optimization:

- This method uses a statistical model to build a belief about the relationships between hyperparameters and model performance.
- It intelligently suggests promising hyperparameter combinations to evaluate, prioritizing regions likely to improve performance.
- More efficient than grid search and random search, but requires more complex setup and expertise.

5. Evolutionary Algorithms:

- These algorithms mimic natural selection principles to iteratively evolve sets of hyperparameter values.
- Each generation undergoes evaluations with the best ones "breeding" to generate new potential hyperparameters.
- Can be powerful for complex search spaces but may require careful configuration and may not always converge to the global optimum.

Additional Considerations:

- Validation set: Use a separate validation set to evaluate the performance of different hyperparameter combinations without overfitting the training data.
- Early stopping: Monitor for performance improvement as you tune hyperparameters and stop the search when it plateaus to avoid overfitting.
- Cross-validation: Repeat the tuning process with multiple folds of your data to ensure your results are robust and generalizable.

Choosing the best hyperparameter tuning method depends on your specific problem, data size, model complexity, and computational resources. Experiment with different approaches and combine them strategically to achieve the best results for your machine learning models.

All paramaters can be tuned

Here are some examples of parameters across different domains to illustrate the breadth of possibilities:

Machine Learning:

- Learning rate
- Number of neurons
- Number of layers
- Activation functions
- Regularization strength
- Batch size
- Epochs

Statistics:

- Mean
- Variance
- Standard deviation
- Correlation coefficient
- Confidence interval
- p-value

Physics:

- Mass
- Velocity
- Acceleration

- Force
- Energy
- Momentum

Engineering:

- Voltage
- Current
- Resistance
- Power
- Temperature
- Pressure

Programming:

- Function arguments
- Variable values
- Return values
- Object properties
- Method calls

And other will find in document of respective packages