

Single logic network using multilayer perceptron neuron model

```
In [1]: import numpy as np
        from sklearn.model_selection import train_test_split
        from sklearn.preprocessing import StandardScaler
        from sklearn.neural_network import MLPClassifier
        from sklearn.datasets import load_iris
        from sklearn.metrics import accuracy_score
```

```
In [2]: iris = load_iris()
        X, y = iris.data, iris.target
```

```
In [3]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
In [4]: scaler = StandardScaler()
        X_train = scaler.fit_transform(X_train)
        X_test = scaler.transform(X_test)
```

```
In [5]: clf = MLPClassifier(hidden_layer_sizes=(64, 32), max_iter=1000, random_state=42)
```

```
In [6]: clf.fit(X_train, y_train)

        y_pred = clf.predict(X_test)
```

```
In [7]: accuracy = accuracy_score(y_test, y_pred)
        print(f"Accuracy: {accuracy:.2f}")
```

Accuracy: 1.00

```
In [ ]:
```