Тестовые вопросы по numpy.mean, numpy.std

1. Функция ***np.mean()*** вычисляет:

2. Функция ***np.std()*** вычисляет:

3. Для вычисления среднего значения элементов вдоль строк двумерного массива ***arr***

следует использовать:

4. Для вычисления стандартного отклонения элементов вдоль столбцов двумерного массива ***arr*** следует использовать:

5. Параметр ***ddof*** в функции ***np.std()*** определяет:

6. Если в массиве есть значения ***NaN***, ***np.mean()*** вернет:

7. Чтобы проигнорировать ***NaN*** при вычислении среднего значения, нужно использовать:

8. Чтобы вычислить взвешенное среднее значение, следует использовать:

Тестовые вопросы по QuantileTransformer

1. QuantileTransformer преобразует данные так, чтобы:

2. Какой параметр QuantileTransformer определяет количество квантилей?

3. Какой параметр QuantileTransformer позволяет выбрать между равномерным и

нормальным распределением преобразованных данных?

4. Если ***output\_distribution='normal'*** , QuantileTransformer преобразует данные так, чтобы они приблизительно соответствовали:

5. QuantileTransformer устойчив к выбросам?

6. Параметр ***random\_state*** в QuantileTransformer используется для:

7. Если в данных есть пропущенные значения (NaN), QuantileTransformer:

Тестовые вопросы по PowerTransformer

1. Основная цель PowerTransformer:

2. PowerTransformer использует следующую технику для преобразования данных:

3. Какой параметр PowerTransformer определяет метод преобразования?

4. Значение ***'yeo-johnson'*** параметра ***method*** означает:

5. Параметр ***standardize*** в PowerTransformer:

6. PowerTransformer подходит для данных:

7. Box-Cox transformation можно применять только к:

Тестовые вопросы по KBinsDiscretizer

1. KBinsDiscretizer используется для:

2. Параметр ***n\_bins*** в KBinsDiscretizer определяет:

3. Какие стратегии разбиения на интервалы поддерживает KBinsDiscretizer (параметр ***strategy***)?

4. Стратегия ***uniform*** в KBinsDiscretizer:

5. Стратегия ***quantile*** в KBinsDiscretizer:

6. Параметр ***encode*** в KBinsDiscretizer определяет способ кодирования преобразованных признаков:

7. ***encode='ordinal'*** в KBinsDiscretizer преобразует данные в:

8. KBinsDiscretizer может обрабатывать пропущенные значения (NaN)?