

ДОКЛАД: "ИСПОЛЬЗОВАНИЕ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ДИАГНОСТИКИ И ПРОГНОЗИРОВАНИЯ КАЧЕСТВА ФУНКЦИОНИРОВАНИЯ БЛОКА УСИЛИТЕЛЯ МОЩНОСТИ

Доклад подготовлен для НИРС
студентом ИВТАСбд-41
Зюзиным Г.А.
Научный руководитель:
Святов К.В.

Цель работы

- Разработать алгоритм и программу для исследования влияния различных факторов на показатели качества **регрессий и классификаций** для обеспечения **диагностики** и **прогнозирования** функционирования радиолокационной аппаратуры.

Актуальность

- Обеспечение надежности функционирования радиолокационной аппаратуры является одной из важнейших задач качественной работы военной техники.
- Особенность рассматриваемой задачи — **малый** объем выборки.
- Прогнозирование мощности и классификация неисправностей на разных частотах (**низкие, средние, высокие**).

Новизна исследования

- Впервые исследовано влияние различных факторов на их эффективность при прогнозировании качества работы блока усилителя мощности.
- Поиск наилучших сочетаний гиперпараметров моделей.

Методы решений. Scikit Learn

- Scikit-learn - один из наиболее широко используемых пакетов Python для Data Science и Machine Learning. Он позволяет выполнять множество операций и предоставляет множество алгоритмов для **нейронных сетей, регрессии и классификации**.



Регрессия. Наборы данных

- Качество функционирования блока усилителя характеризуется выходной мощностью на различных частотах и зависит от характеристик поставляемых **клистронов**.
- Наборы данных содержат по 90 строк для каждой частоты, т.е. для каждой частоты **своя модель**.
- Целевая переменная - **выходная мощность** ун, ус, ув с блока в различных точках частотного диапазона, [кВт].
- $x_1 - x_{10}$ – предикаты, которые обозначают следующие характеристики клистрона:

Характеристики клистрона	
x_1 – импульсный ток катода [А]	x_6 – потребляемая мощность [кВт]
x_2 – напряжение накала [В]	x_7 – входная оптимальная мощность [мВт]
x_3 – ток накала [А]	x_8 – выходная мощность с клистрона [кВт]
x_4 – токопрохождение в статике [%]	x_9 – импульсный ток резонаторного блока [А]
x_5 – ток управляющего электрода [мА]	x_{10} – коэффициент усиления

Регрессия. Предобработка данных

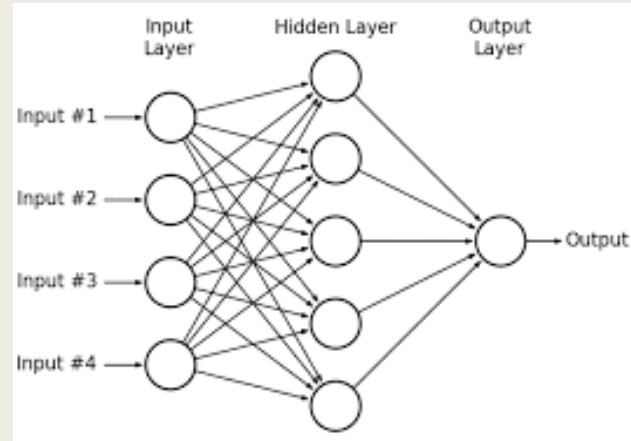
- Разделение выборки: 80% – обучающая, 20% – тестовая.
- Стандартизация (`StandardScaler()`).
- Стандартизация наборов данных является **общим требованием для многих моделей** машинного обучения, реализованных в `scikit-learn`; они могут вести себя плохо, если отдельные признаки не будут более или менее похожи на стандартные нормально распределенные данные: гауссовы с **нулевым средним и единичной дисперсией**.

$$X_{scaled} = \frac{X - \mu}{\sigma}$$

- где,
 - μ – математическое ожидание;
 - σ – стандартное отклонение.

Регрессия. Нейронная сеть

- **MLPRegressor** — это модель нейросети для регрессии
- **MLP (Multi-Layer Perceptron)** — многослойный перцептрон.
- Используется для предсказания численных значений.
- Состоит из входного слоя, скрытых слоев и выходного слоя.



- Параметры вроде **hidden_layer_sizes**, **activation**, **alpha** и др. управляют её поведением.

Нейронная сеть

- **Многослойный перцептрон** (Multilayer Perceptron - MLP) - это алгоритм контролируемого обучения, который обучает функцию $f(\cdot): R^m \rightarrow R^o$ путем обучения на наборе данных, где m - количество размерностей для входа, а o - количество размерностей для выхода. Есть набор признаков $X = x_1, x_2, \dots, x_m$ и цель y .
- В scikit learn можно варьировать следующие **основные параметры** нейронной сети:
 - **hidden_layer_sizes**. *Скрытые слои и их количество;*
 - **activation**. *Функция активации скрытого слоя;*
 - **solver**. *Алгоритм оптимизации весов;*
 - **alpha**. *Регуляризация L2;*
 - и другие.
- Существует много других **дополнительных параметров**, которые можно варьировать, исходя из **основных параметров**. Например, **momentum** (Импульс для обновления градиентного спуска) (при `solver='sgd'`), **epsilon** (Значение для численной стабильности в adam (при `solver='adam'`)) и другие.

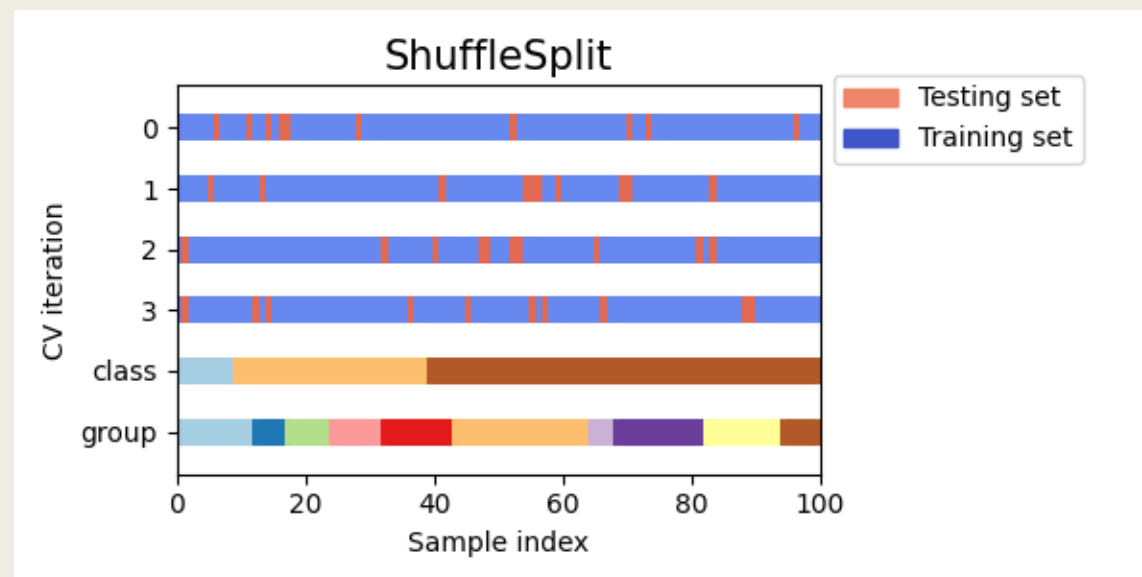
Регрессия. Сетка гиперпараметров (Grid Search)

- Сетка с различными параметрами, из которых составляются разные комбинации параметров модели, чтобы найти наилучшую конфигурацию.
- Это делается автоматически, модель обучается на каждой комбинации.

```
# Определение параметров для поиска по сетке
param_grid = {
    'hidden_layer_sizes': [(8,), (10,), (12,)],
    'activation': ['logistic'],
    'solver': ['adam'],
    'alpha': [0.92, 0.96, 1.0],
    'beta_1': [0.8, 0.85, 0.9],
    'beta_2': [0.9, 0.95, 0.99],
    'early_stopping': [True],
    'learning_rate_init': [0.03, 0.04, 0.05],
    'tol': [0.01, 0.05, 0.1],
    'epsilon': [1e-6],
    'max_iter': [200],
    'n_iter_no_change': [5, 10],
    'validation_fraction': [0.1, 0.2, 0.3]
}
```

Регрессия. Кросс-валидация (ShuffleSplit)

- Кросс-валидация – м-д оценки модели, при котором данные делятся на **фолды** (folds), и модель многократно обучается на одних и проверяется на других.
- **ShuffleSplit** – это тип кросс-валидации, где:
 - данные перемешиваются случайно;
 - **обучающая выборка (80%)** делится на обучающую и валидационную, **тестовая выборка (20%)** будет использована для подсчёта характеристик



Регрессия. GridSearchCV

- GridSearchCV — это поиск по сетке с перекрёстной проверкой. GridSearch — "поиск по сетке" (всех вариантов параметров), CV (Cross-Validation) — "кросс-валидация",
- Что делает:
 - Перебирает разные комбинации параметров модели и оценивает каждую с помощью кросс-валидации.
 - Выбирает по заданному параметру, в данном случае MAPE (*neg_mean_absolute_percentage_error*).

Регрессия. MAPE и RMSE

- Модель ищется по `neg_mean_absolute_percentage_error` - это отрицательная средняя абсолютная процентная ошибка (Negative Mean Absolute Percentage Error).
- **MAPE** – это средняя процентная ошибка:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|Y_i - \hat{Y}_i|}{Y_i}$$

- Корень средней квадратичной ошибки (**RMSE**) является одним из основных показателей эффективности для модели прогнозирования регрессии, показывает среднюю разницу между значениями, спрогнозированными и фактическими значениями.

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}}$$

Классификация. Наборы данных

- Состояние усилителя блока были разделены и охарактеризованы на три класса для **каждого диапазона частот**:
 - $Z=1$ – хорошее;
 - $Z=2$ – удовлетворительное;
 - $Z=3$ - неудовлетворительное.
- Т.е. для каждого диапазона своя модель классификации. Наборы данных содержат по 100 строк для каждого диапазона частот.
- Целевая переменная – Z_n , Z_c или Z_v (в зависимости от диапазона).

Предикторы для классификации

Y_n, Y_c, Y_v – выходная мощность функционального блока [кВт]

U – напряжение питания по высоковольтной линии [кВ]

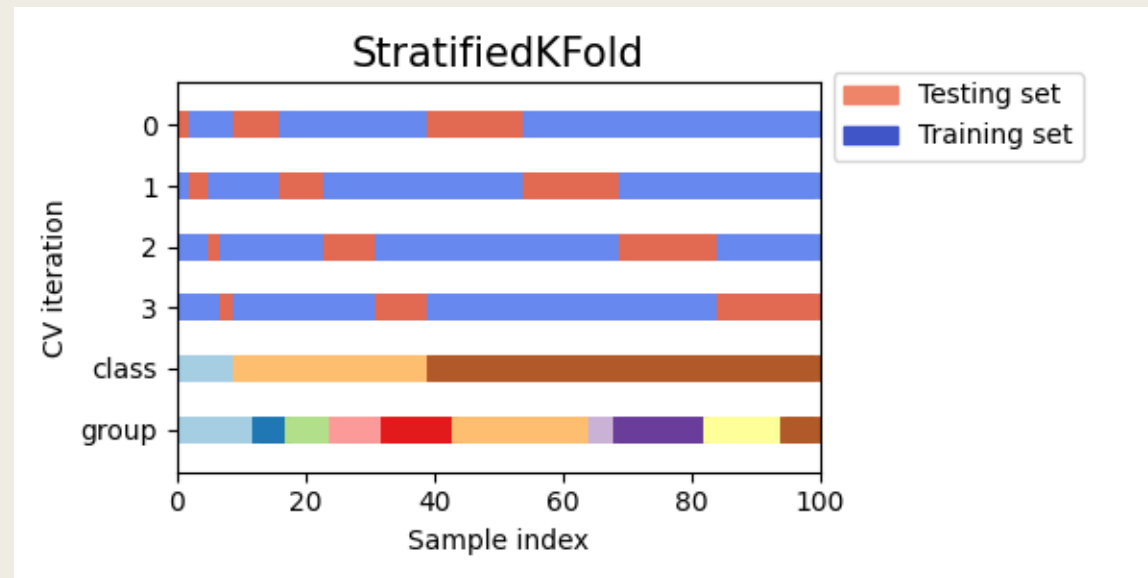
T – температура внешнего воздействия [$^{\circ}\text{C}$]

Классификация. Набор решений

- Во многом по аналогии с регрессии. Разделение выборки на **обучающую** и **тестовую** 75% и 25%, соответственно.
- Отличия: **MLPClassifier**, **SMOTE**, **StratifiedKFold**, **f1_macro**.
 - ***MLPClassifier***
 - Модель нейросети для классификации (предсказание категорий, а не чисел). Аналог **MLPRegressor**, но для задач типа "к какому классу относится".
 - ***SMOTE (Synthetic Minority Over-sampling Technique)***
 - Используется, если классы **несбалансированы** (например, одного класса сильно меньше).
 - Создаёт искусственные примеры меньшинства, чтобы выровнять баланс, то есть помогает избежать перекоса модели в сторону "большого" класса.

Классификация. Кросс-валидация

- **StratifiedKFold** — это тип кросс-валидации, где:
 - Данные делятся на K частей.
 - В каждом фолде сохраняется **та же пропорция классов**, как и во всём наборе данных.
 - Полезен при несбалансированных классах, чтобы модель училась и проверялась на равных условиях.



Классификация. F1_macro

- Модель ищется по **f1_macro** - это о среднее гармоническое между точностью (**precision**) и полнотой (**recall**).

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

- Что такое **f1_macro**?
 - F1 рассчитывается отдельно для каждого класса,
 - после берётся среднее значение по всем классам.

$$F1_{macro} = \frac{1}{N} \sum F1_i$$

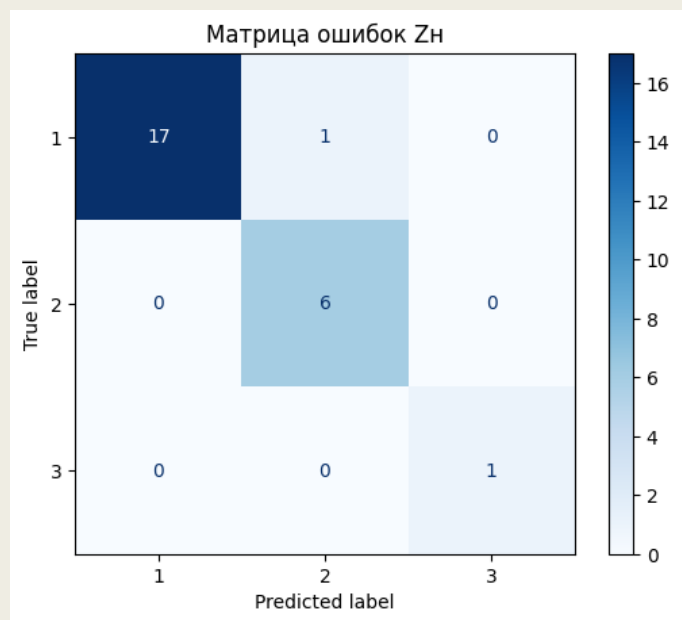
где,

- N — количество классов,
- $F1_i$ — F1-мера для i -го класса

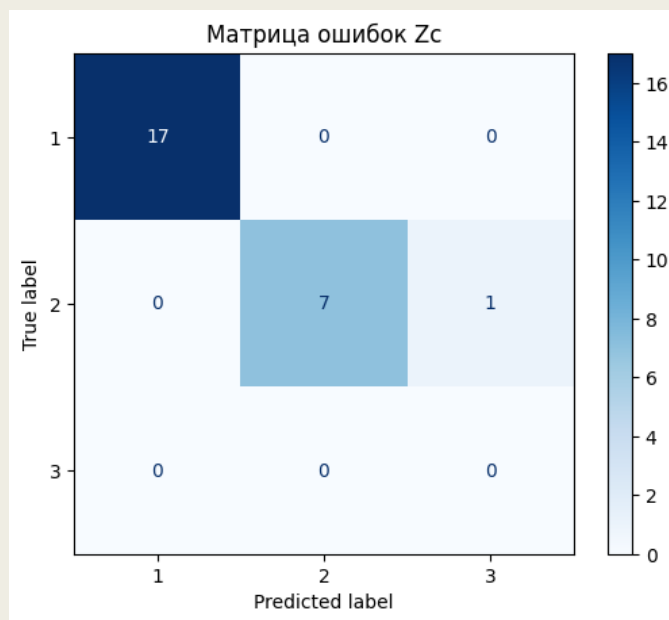
Результаты. Регрессия

	Значения на тестовой выборке	Значения на кросс-валидации
Y _н	MAPE = 8.92% RMSE = 3.24	MAPE = 8,91%
Y _в	MAPE = 7.98% RMSE = 2.47	MAPE = 8.01%
Y _с	MAPE = 8.13% RMSE = 2.35	MAPE = 8.41 %

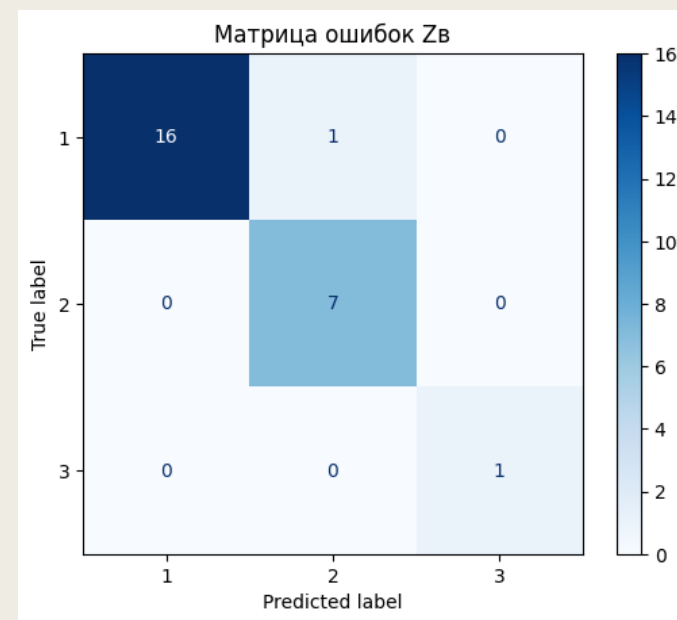
Результаты. Классификация



F-мера (Zн): 0.9610



F-мера (Zс): 0.9787



F-мера (Zв): 0.9607

Перспективы

- Публикация статьи на тему: «Нейронная сеть для оценки качества функционирования блока усилителя мощности» с соавтором В.О. Кушнаревым (АО Механический завод), как постановщиком задачи.
- Защита выпускной квалифицированной работы по этим задачам.
- Введение моделей в эксплуатацию.

Полученные результаты – регрессия по MAPE и классификация по F-мере – удовлетворяют требованиям заказчика.