

Название: Исследование и оптимизация радиально-базисных сетей для анализа данных CS:GO Professional Matches.

Цель работы:

1. Изучить принципы работы RBF-сетей, их преимущества и недостатки. (2 балла)
2. Применить RBF-сети для решения задачи классификации или регрессии на датасете "CS:GO Professional Matches". (2 балла)
3. Исследовать влияние "проклятия размерности" на производительность RBF-сетей. (3 балла)
4. Оптимизировать структуру и параметры RBF-сетей для повышения точности и масштабируемости. (3 балла)
5. Сравнить производительность различных RBF-архитектур и методов обучения. (1 или 2 балла)

Датасет: "CS:GO Professional Matches" (<https://www.kaggle.com/datasets/gabrieltardochi/counter-strike-global-offensive-matches>)

План работы:

1. Подготовка данных и предварительный анализ (2 часа):
 - Загрузка и осмотр данных:
 - Загрузить датасет.
 - Определить количество строк и столбцов.
 - Вывести первые несколько строк для ознакомления.
 - Предоставить краткое описание датасета (что представляют собой столбцы).
 - Очистка данных (при необходимости):
 - Обработать пропущенные значения (удаление или заполнение).
 - Удалить дубликаты.
 - Преобразовать категориальные признаки в числовые (one-hot encoding или label encoding).
 - Разведочный анализ данных (EDA):
 - Визуализировать распределения важных признаков (гистограммы, boxplots).
 - Рассчитать корреляционные матрицы.
 - Определить целевую переменную (например, Team Performance Metric и т.д.). Обосновать выбор целевой переменной.
 - Выделить признаки, которые кажутся наиболее важными для целевой переменной.
 - Разделение данных:
 - Разделить датасет на обучающую, валидационную и тестовую выборки (например, 70/15/15).
2. Реализация и обучение базовой RBF-сети (4 часа):
 - Выбор библиотек:
 - PyTorch или TensorFlow (предпочтительнее PyTorch, поскольку примеры кода из лекции на нем).
 - Scikit-learn для вспомогательных задач (например, k-means).
 - Реализация RBF-слоя:
 - Реализовать RBF-слой с использованием выбранной библиотеки. Обязательно реализовать несколько видов RBF (Гауссова, мультиквадратичная и т.д.).
 - Функции для вычисления расстояний (евклидово расстояние).

- Инициализация центров:
 - Реализовать метод k-средних для определения центров RBF-функций (использовать Scikit-learn или реализовать самостоятельно).
 - Реализовать метод случайного выбора центров.
 - Сравнить результаты обучения с разными методами инициализации.
- Вычисление ширины RBF:
 - Реализовать правило ближайшего соседа для вычисления ширины σ .
 - Реализовать глобальный метод вычисления ширины.
 - Сравнить результаты обучения с разными методами вычисления ширины.
- Обучение выходного слоя:
 - Реализовать обучение выходного слоя с помощью линейной регрессии (через псевдообратную матрицу).
 - Реализовать обучение выходного слоя с помощью градиентного спуска (различные варианты: пакетный, стохастический, мини-батч).
 - Сравнить скорость и сходимость разных методов обучения.
- Оценка производительности:
 - Вычислить метрики производительности на валидационной выборке (точность, precision, recall, F1-score для классификации; RMSE, MAE для регрессии).
 - Проанализировать полученные результаты.

3. Исследование "проклятия размерности" (3 часа):

- Увеличение размерности данных:
 - Выбрать подмножество признаков (например, 5, 10, 15, 20).
 - Обучить RBF-сеть для каждого подмножества признаков.
 - Зафиксировать количество центров.
 - Сравнить производительность (метрики, время обучения) с ростом размерности.
- Анализ зависимости от числа центров:
 - Для фиксированной размерности данных обучить RBF-сети с разным количеством центров (например, 10, 20, 50, 100).
 - Оценить влияние количества центров на производительность и время обучения.
 - Визуализировать зависимость производительности от размерности и числа центров.
- Выводы:
 - Определить, проявляется ли "проклятие размерности" в данном датасете.
 - Оценить, насколько сильно увеличение размерности влияет на производительность RBF-сети.

4. Оптимизация RBF-сети (4 часа):

- RBF-сеть на основе многослойного персептрона (MLP):
- Реализовать гибридную модель: RBF-слой + MLP.
- Обучить модель с использованием градиентного спуска.
- Подобрать параметры MLP (количество слоев, количество нейронов в слоях, функция активации).
- Методы ускорения обучения:
 - Реализовать mini-batch k-means для кластеризации.

- Использовать KD-деревья для ускорения поиска ближайших центров.
- Реализовать рандомизированный SVD для ускорения псевдообращения.
- Сравнить время обучения с использованием различных методов ускорения.
- Оптимизация гиперпараметров:
 - Использовать GridSearch или RandomSearch для поиска оптимальных значений гиперпараметров (количество центров, ширина σ , параметры регуляризации, параметры MLP).
- Ансамбли RBF-сетей (компенсирует три невыполненных подпункта помеченных *):
 - Реализовать bagging RBF-сетей (обучение нескольких моделей с разными подмножествами данных и усреднение предсказаний).
 - Сравнить производительность ансамбля с производительностью одиночной RBF-сети.

5. Сравнение с другими моделями (2 часа):

- Обучение других моделей:
 - Обучить другие модели машинного обучения на том же датасете (например, логистическую регрессию, SVM, случайный лес, градиентный бустинг).
- Сравнение производительности:
 - Сравнить производительность RBF-сетей с другими моделями на тестовой выборке.
 - Оценить преимущества и недостатки RBF-сетей по сравнению с другими моделями.

6. Анализ результатов и выводы (3 часа):

- Анализ:
 - Подробный анализ полученных результатов.
 - Сравнение различных архитектур RBF-сетей и методов обучения.
 - Обсуждение преимуществ и недостатков RBF-сетей для данной задачи.
 - Интерпретация полученных моделей (например, какие признаки наиболее важны для предсказания победителя матча).
- Выводы:
 - Формулировка основных выводов по результатам работы.
 - Обсуждение перспектив дальнейших исследований.
 - Оценка применимости RBF-сетей для решения задач анализа данных CS:GO.

Критерии оценки:

- Полнота выполнения работы.
- Правильность реализации RBF-сетей и других моделей.
- Качество анализа данных и результатов экспериментов.
- Самостоятельность выполнения работы^_^

Телега:

@SaTmOrF