ЗАДАЧА 1: Да се направи скрипт, който чрез НМ, да **реализира класификация**.

Преформатиране на входните данни не се налага, защото те са генерирани нормализирани с медиана 0 и стандартно отклонение 0.1. Данни за никой атрибут не липсват и освен това има незначителен шум

Таблица 1: Описание на формата на подаваните данни в мрежата[[1]](#footnote-1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Име на полето (колоната) от мрежата | Тип на данните в полето | Тип на полето | Изчисляемо | Пояснение |
| Х координата | Real | Входно | Не | 0,15 |
| У координата | Real | Входно | Не | 0,25 |
| Class | Boolean | Изходно | Не | True |

Таблица 2: Първоначална архитектура на мрежата

|  |  |  |
| --- | --- | --- |
| Слой неврони | Брой неврони в слоя преди оптимизацията | Брой неврони в слоя след оптимизацията |
| Входни | 2 | 2 |
| Изходни | 1 | 1 |
| Скрит слой 1 | 0 | 4 |

Таблица 3: Параметри на мрежата

|  |  |  |
| --- | --- | --- |
| Параметри на мрежата | преди оптимизацията | След оптимизацията |
| Активационна функция | Sigmoid function | Sigmoid function |
| Learning rate | 0.5 | 0.7 |
| Momentum | 0 | 0.8 |

Оптимизация чрез на НМ (ако е използван) – посочете в свободен методиката на оптимизация

Таблица 4: Крайни резултати за задачата

|  |  |
| --- | --- |
| Параметър за оценка | Стойност |
| Общ брой записи | 20 000 |
| Брой записи в тренинг сета | 16 000 |
| Брой записи за валидация | 1 000 |
| Брой записи за реално тестване | 3 000 |
| Брой цикли на трениране на мрежата | 51 |
| Избран критерий за стоп на тренинга | Всички грешки са под 0.01 или 100% от валидационното множество е правилно след закръгляне |
| Средна Реално измерена грешка върху три произволни примера | 0,00000375 |
| Време за трениране на мрежата | 15 секунди |
| Време за оптимизация на архитектурата[[2]](#footnote-2) | 10 секунди |
| Време за зареждане на данните | 2 секунди |
| Общо време | 27 секунди |

ЗАДАЧА 2: Да се направи скрипт, който чрез НМ, да **реализира прогноза**.

Данните в текущия си формат не са подходящи за регресия. Затова ще се опитаме да познаем точка по зададени две преди нея по някоя от кривите. Това означава, че използваме същите данни, но на мрежата ще подадем три точки по кривата (сортирани) и ще очакваме да върне четвърта точка отново по същата крива.

Таблица 1: Описание на формата на подаваните данни в мрежата[[3]](#footnote-3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Име на полето (колоната) от мрежата | Тип на данните в полето | Тип на полето | Изчисляемо | Пояснение |
| X1 координата | Real | Входно | Не | 0,15 |
| У1 координата | Real | Входно | Не | 0,25 |
| Х2 координата | Real | Входно | Не | 0,15 |
| У2 координата | Real | Входно | Не | 0,25 |
| Хtarget координата | Real | Изходно | Не | 0,15 |
| Уtarget координата | Real | Изходно | Не | 0,25 |

Таблица 2: Първоначална архитектура на мрежата

|  |  |  |
| --- | --- | --- |
| Слой неврони | Брой неврони в слоя преди оптимизацията | Брой неврони в слоя след оптимизацията |
| Входни | 4 | 4 |
| Изходни | 2 | 2 |
| Скрит слой 1 | 0 | 6 |

Таблица 3: Параметри на мрежата

|  |  |  |
| --- | --- | --- |
| Параметри на мрежата | преди оптимизацията | След оптимизацията |
| Активационна функция | Sigmoid function | Sigmoid function |
| Learning rate | 0.5 | 0.7 |
| Momentum | 0 | 0.8 |

Оптимизация чрез на НМ (ако е използван) – посочете в свободен методиката на оптимизация

Таблица 4: Крайни резултати за задачата

|  |  |
| --- | --- |
| Параметър за оценка | Стойност |
| Общ брой записи | 14 000 |
| Брой записи в тренинг сета | 10 900 |
| Брой записи за валидация | 1 000 |
| Брой записи за реално тестване | 2 100 |
| Брой цикли на трениране на мрежата | 80 |
| Избран критерий за стоп на тренинга | Когато всички грешки станат по-малки от 0.012 |
| Средна Реално измерена грешка върху три произволни примера | 0.000346 |
| Време за трениране на мрежата | 12 секунди |
| Време за оптимизация на архитектурата[[4]](#footnote-4) | 3 секунди |
| Време за зареждане на данните | 1 секунда |
| Общо време | 16 секунди |

ЗАДАЧА 3: Да се направи скрипт, който чрез НМ, да **реализира клъстериазация**.

Не е необходима обработка на данните

Таблица 1: Описание на формата на подаваните данни в мрежата[[5]](#footnote-5)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Име на полето (колоната) от мрежата | Тип на данните в полето | Тип на полето | Изчисляемо | Пояснение |
| Х координата | Real | Входно | Не | 0,15 |
| У координата | Real | Входно | Не | 0,25 |

Клъстеризирането на данните е направено със Self-organizing map. При този алгоритъм няма оптимизиране на мрежата. Има само един слой с толкова неврона колкото са клъстърите.

1. Попълнените данни са примерни [↑](#footnote-ref-1)
2. Може да бъде 0, ако не е правена оптимизация на мрежата. [↑](#footnote-ref-2)
3. Попълнените данни са примерни [↑](#footnote-ref-3)
4. Може да бъде 0, ако не е правена оптимизация на мрежата. [↑](#footnote-ref-4)
5. Попълнените данни са примерни [↑](#footnote-ref-5)