

Упражнение 9 (Избор на 1 от 2 задачи). Задача 1 не изисква програмна реализация, за разлика от задача 2, при която се изисква програмна реализация.

Задача 1.

Конструирайте персептрон с два входа, който имплементира булевата функция $A \wedge \neg B$ (напомням, че булевите стойности се кодират като 1 (истина) и -1 (лъжа)).

Да се намерят теглата на мрежата.

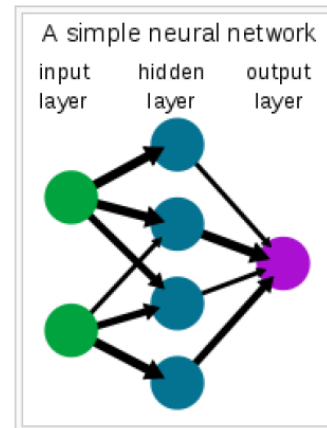
Задача 2. Backpropagation

Задача за Невронна Мрежа:

Да се реализира невронна мрежа самообучаваща се чрез Backpropagation.

При реализацията на невронната мрежа:

- Входните неврони са равни на броя на атрибутите, с които се описват примерите
- Изходните възли са равни на броя на броя на класовете
- Броя на невроните от скрития слой е параметър на системата



При реализацията на невронната мрежа първо инициализираме всички връзки м/у възлите с произволни стойности от интервала $(-0.05, 0.05)$.

След това правим обучение на мрежата използвайки всички примери K пъти (K е параметър на системата)

- За всеки обучаващ пример изчисли активацията в скрития и изходния слой посредством формулата $x_i = \frac{1}{1 + e^{-sum}}$, където $sum = \sum w_{ji} x_j$; w_{ji} - връзка между x_i ; x_j - възел от предишния слой
- Използва се обратно разпространение на грешката (backpropagation) за подобряване на теглата на невронната мрежа.

- За всеки възел изчисляваме грешка, която е равна на:

$$\delta_k \leftarrow o_k (1 - o_k) (t_k - o_k)$$

- За изходен възел: δ_k , където O_k е получената стойност във възела а T_k е търсената

$$\delta_h \leftarrow o_h (1 - o_h) \cdot \sum_{k \in \text{изходни}} w_{kh} \delta_k$$

- За всеки скрит възел:
- Обновяваме теглата използвайки изчислените грешки. За всяко тегло от мрежата: $w_{ij} = w_{ij} + \mu \cdot \delta_j \cdot x_i$, където μ - константа между 0 и 1 (например 0.1), δ_j - грешката която се е получила в края на реброто, x_i - активността на възела в началото на реброто.

Приятна работа ☺