perceptron-demo.ipynb

File Edit View Insert Runtime Tools Help Last saved at March 3

Comment Share

+ Code + Text

Connect Colab AI

Start coding or generate with AI.

Double-click (or enter) to edit

What is a Perceptron?

Perceptron is a fundamental unit of a neural network, essentially functioning as an algorithm for supervised machine learning.

It consists of several components:

- **Inputs (x1, x2, x3, ...)**: These are the features fed into the algorithm.
- **Weights (W1, W2, W3, ...)**: Each input is associated with a weight, signifying the importance or influence of that input.
- **Summation Block:** This component sums up the product of inputs and their respective weights.
- **Bias (B):** An additional parameter that influences the output independently of the inputs.
- **Activation Function:** A function that transforms the sum into an output, often aiming to fit it within a certain range.
- The perceptron computes an **output (y)** using the formula:

> Output (y) = Activation Function(∑(Inputs×Weights)+Bias)

> Output (y) = Activation Function([(x1×W1) + (x2×W2) + (x3×W3) + Bias)])

Real-Life Problem: Predicting Students placement

To demonstrate the perceptron's application, consider predicting Student placement on features like cgpa and resume_score.

The perceptron model uses these features to predict whether a students take placement or nnot.

Features:

cgpa (x1)

resume_score (x2)

Label:

Placement (y): Indicates if a placement was made (1) or not (0).

```
[ ] import numpy as np
    import pandas as pd
    import seaborn as sns
    import matplotlib.pyplot as plt
```

```
[ ] from google.colab import drive
    drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
df = pd.read_csv('/content/drive/My Drive/Colab Notebooks/Dataset/placement.csv')
```

```
[ ] print(df.shape)
    df.head()
```

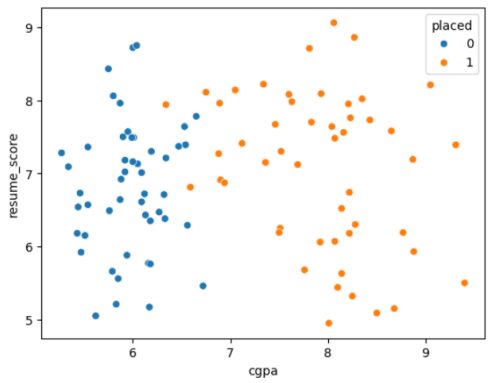
```
(100, 3)
```

	cgpa	resume_score	placed
0	8.14	8.53	1

0	8.14	6.52	1
1	6.17	5.17	0
2	8.27	8.86	1
3	6.88	7.27	1
4	7.52	7.30	1

```
sns.scatterplot(df['cgpa'],df['resume_score'],hue=df['placed'])
#sns.scatterplot(df, x="cgpa", y="resume_score", hue="placed")
```

```
<Axes: xlabel='cgpa', ylabel='resume_score'>
```




```
X = df.iloc[:,0:2]
y = df.iloc[:,-1]
```

```
Start coding or generate with AI.
```

```
from sklearn.linear_model import Perceptron
p = Perceptron()
```

```
p.fit(X,y)
```

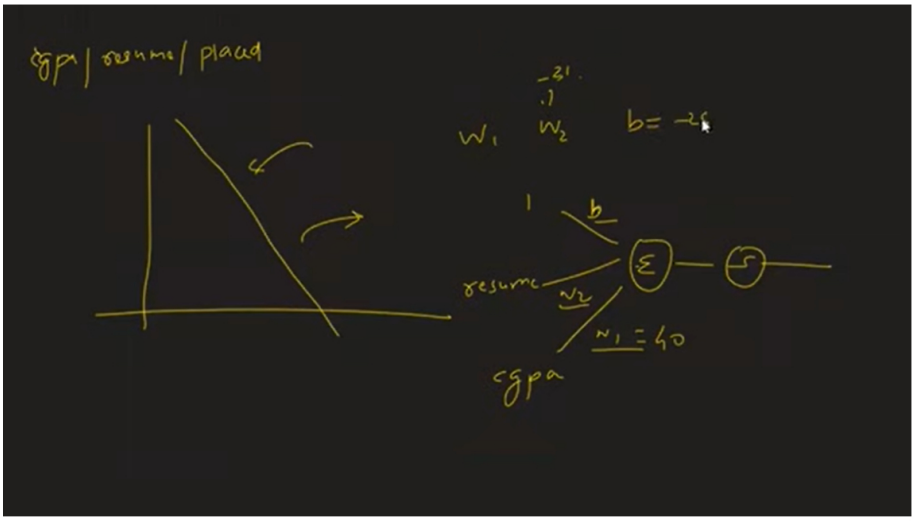
```
Perceptron()
```

```
p.coef_
```

```
array([[ 40.26, -36.  ]])
```

```
p.intercept_
```

```
array([-25.])
```

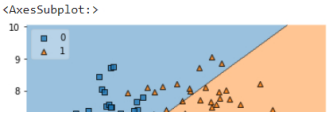


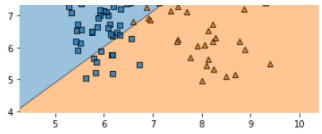
Double-click (or enter) to edit

```
from mlxtend.plotting import plot_decision_regions
```

```
plot_decision_regions(X.values, y.values, clf=p, legend=2)
```

```
<AxesSubplot:>
```






```
[ ] Start coding or generate with AI.
```

https://playground.tensorflow.org/