Chapter10-LogisticRegression

January 7, 2020

1 Logistic Regression from a Neural Networks Perspective

```
In [1]: import numpy as np
        import matplotlib.pyplot as plt
```

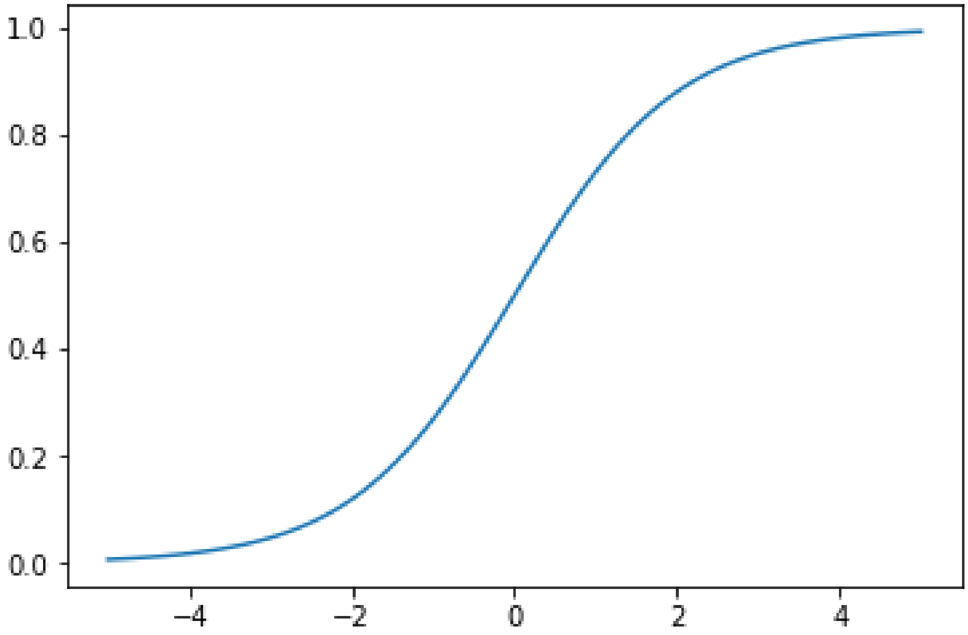
1.1 Sigmoid Activation Function

```
In [4]: x = np.linspace(-5, 5, 100)

        def sigmoid(z):
            return 1 / (1 + np.exp(-z))

        s = sigmoid(x)
        plt.plot(x, s)

Out[4]: [<matplotlib.lines.Line2D at 0x7f0ac3eafe10>]
```



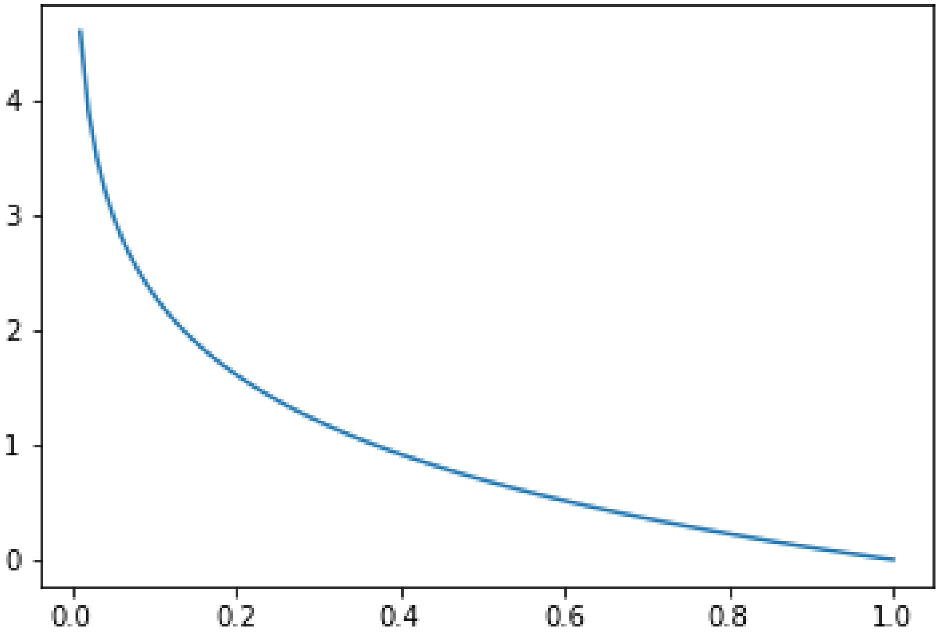
1.2 Cross-entropy Loss Function

```
In [5]: x = np.linspace(0, 1, 100)
        Ly1 = - np.log(x) # Loss if the true y is 1
```

```
/home/joerg/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:2: RuntimeWarning: div
```

```
In [6]: plt.plot(x, Ly1)
```

```
Out[6]: [<matplotlib.lines.Line2D at 0x7f0ac3e96588>]
```

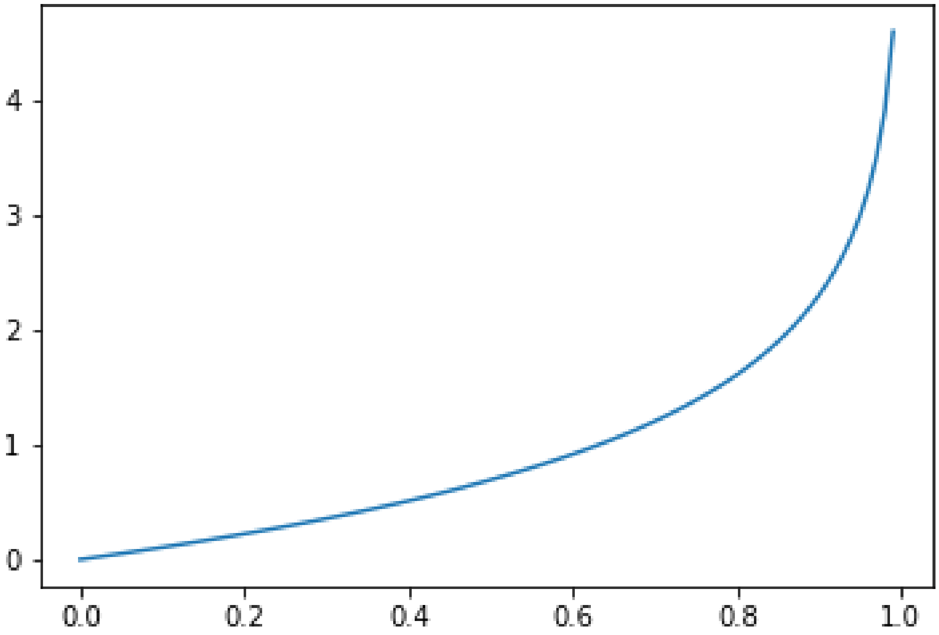


```
In [7]: Ly0 = - np.log(1-x) # Loss if the true y is 0
```

```
/home/joerg/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: RuntimeWarning: div
  """Entry point for launching an IPython kernel.
```

```
In [8]: plt.plot(x, Ly0)
```

```
Out[8]: [<matplotlib.lines.Line2D at 0x7f0ac3debef0>]
```



```
In [ ]:
```