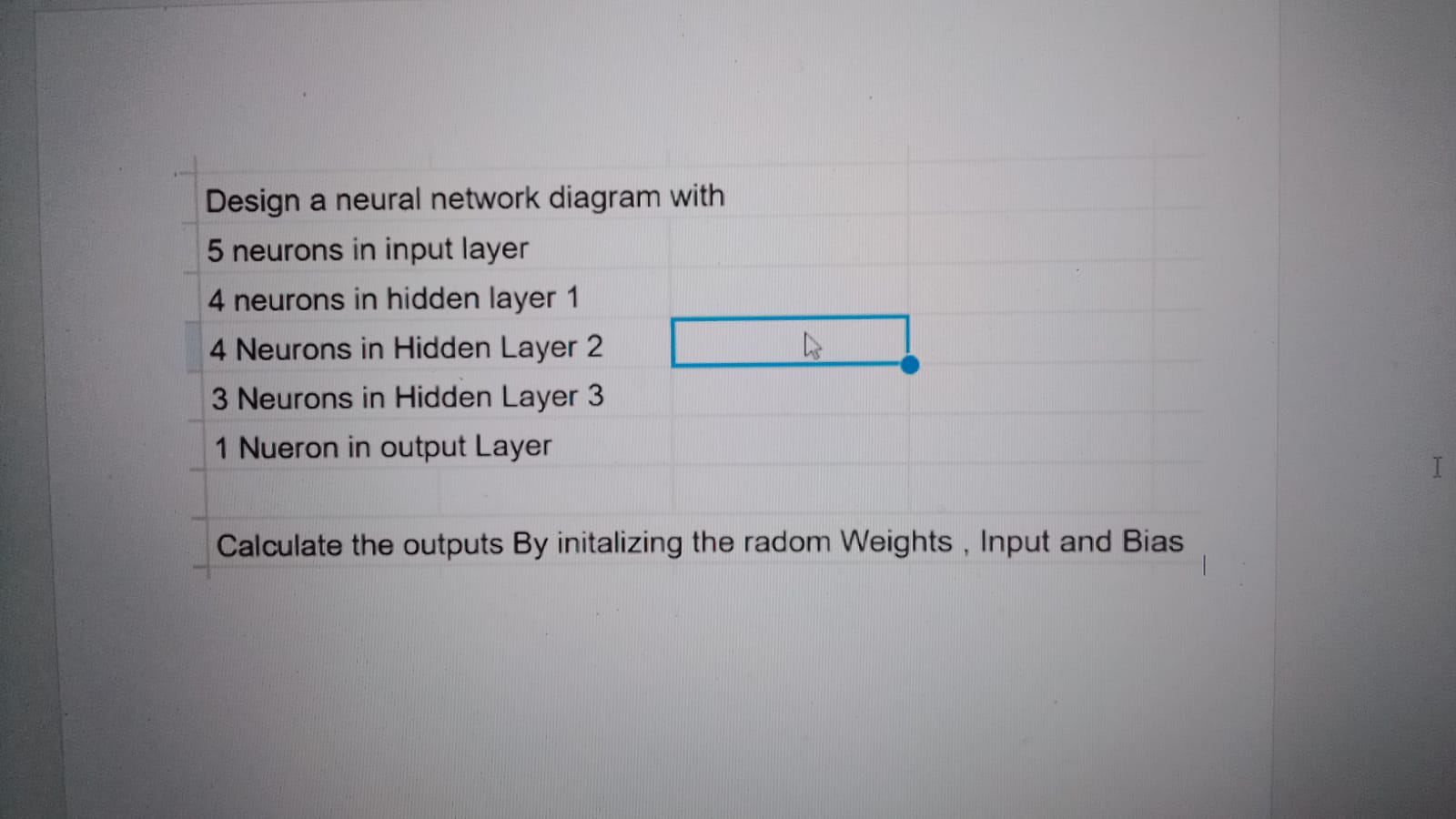
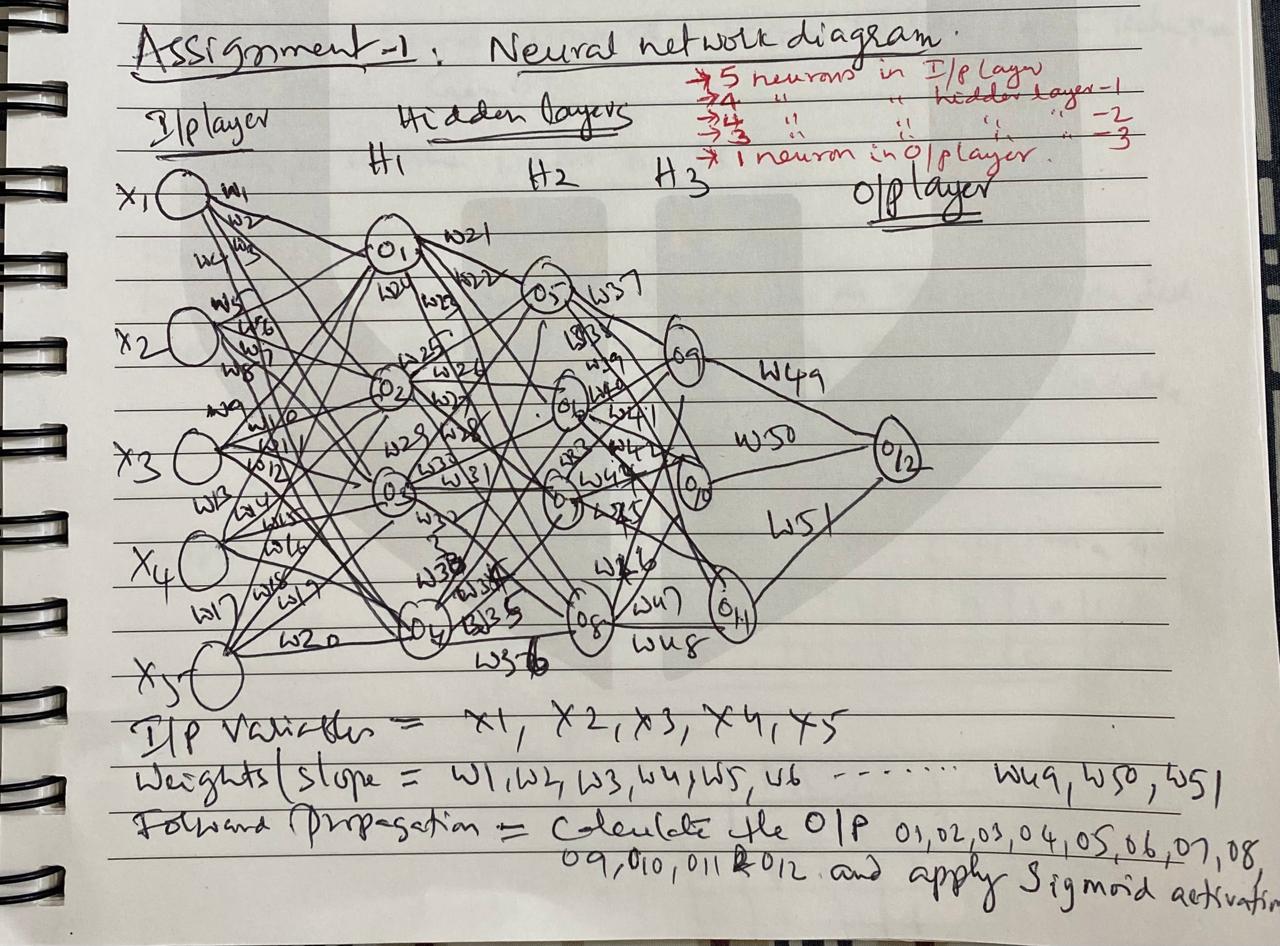
**IC Assignment -1**





Let's go **step by step** for the **entire forward propagation**.

**Step 1: Understanding the Neural Network Structure**

1. **Input Layer (5 neurons):**
   * x1,x2,x3,x4,x5
2. **Hidden Layer 1 (4 neurons):**
   * o1,o2,o3,o4
3. **Hidden Layer 2 (4 neurons):**
   * o5,o6,o7,o8 ​
4. **Hidden Layer 3 (3 neurons):**
   * o9,o10,o11
5. **Output Layer (1 neuron):**
   * o12 ​

**Weights and Inputs:**

* **Weights:** w1,w2,...,w51
* **Weight values:** w1=10,w2=20,...,w51=510 (incrementing by 10)
* **Bias:** b=0.1b = 0.1
* **Inputs:** x1=10,x2=20,x3=30,x4=40,x5=50

**Step 2: Compute Hidden Layer 1 (o1,o2,o3,o4​)**

* Each neuron receives inputs from **all 5 input neurons**.
* **Calculate o1 ​**

o1=x1w1+x2w5+x3w9+x4w13+x5w17+b o1=(10×10)+(20×50)+(30×90)+(40×130)+(50×170)+0.1 o1=100+1000+2700+5200+8500+0.1

o1=17500.1

* **Calculate o2**

o2=x1w2+x2w6+x3w10+x4w14+x5w18+b o2=(10×20)+(20×60)+(30×100)+(40×140)+(50×180)+0.1 o2=200+1200+3000+5600+9000+0.1

o2=19000.1

* **Calculate o3**

o3=x1w3+x2w7+x3w11+x4w15+x5w19+b o3=(10×30)+(20×70)+(30×110)+(40×150)+(50×190)+0.1

o3=300+1400+3300+6000+9500+0.1

o3=20700.1

* **Calculate o4**

o4=x1w4+x2w8+x3w12+x4w16+x5w20+b o4=(10×40)+(20×80)+(30×120)+(40×160)+(50×200)+0.1 o4=400+1600+3600+6400+10000+0.1

o4=22400.1

**Step 3: Compute Hidden Layer 2 (o5,o6,o7,o8​)**

Each neuron receives input from o1,o2,o3,o4​.

**Calculate o5​**

o5=o1w21+o2w25+o3w29+o4w33+b o5=(17500.1×210)+(19000.1×250)+(20700.1×290)+(22400.1×330)+0.1 =3675021+4750025+6003005+7392005+0.1

o5=21840056.1

**Calculate o6 ​**

**o6​=o1​w22​+o2​w26​+o3​w30​+o4​w34​+b**

**o6=(17500.1×220)+(19000.1×260)+(20700.1×300)+(22400.1×340)+0.1**

**o6=3850022+4940026+6210030+7616005+0.1**

**o6=22680083.1**

**Calculate o7**

**o7​=o1​w23​+o2​w27​+o3​w31​+o4​w35​+b**

**o7**=**(17500.1×230)+(19000.1×270)+(20700.1×310)+(22400.1×350)+0.1**

**o7=4025023+5130027+6417031+7840005+0.1**

**o7=23520106.1**

**Calculate o8**

**o8​=o1​w24​+o2​w28​+o3​w32​+o4​w36​+b**

**o8=(17500.1×240)+(19000.1×280)+(20700.1×320)+(22400.1×360)+0.1**

**o8=4200024+5320028+6624032+8064005+0.1**

**o8=24360120.1**

**Step 4: Apply Sigmoid Activation to o1​ to o8**

**Since all values are very large, their sigmoid outputs will be 1.0:**

**σ(x) = 1/1+e−x**

**For large x, e−x approaches 0, making:**

**σ(x)≈1.0**

**Thus:**

**o1​=1.0, o2​=1.0, o3​=1.0, o4​=1.0, o5​=1.0, o6​=1.0, o7​=1.0, o8​=1.0**

**Step 5: Compute Hidden Layer 3 (o9,o10,o11)**

**Each neuron in Hidden Layer 3 receives input from o5,o6,o7,o8​.**

**Calculate o9​:**

**o9​=(o5​×w37​)+(o6​×w41​)+(o7​×w45​)+(o8​×w49​)+b**

**o9=(1.0×370)+(1.0×410)+(1.0×450)+(1.0×490)+0.1**

**o9=370+410+450+490+0.1**

**o9=1720.1**

**Calculate o10:**

**o10​=(o5​×w38​)+(o6​×w42​)+(o7​×w46​)+(o8​×w50​)+b**

**o10=(1.0×380)+(1.0×420)+(1.0×460)+(1.0×500)+0.1**

**o10=380+420+460+500+0.1**

**o10=1760.1**

**Calculate o11:**

**o11​=(o5​×w39​)+(o6​×w43​)+(o7​×w47​)+(o8​×w51​)+b**

**o11=(1.0×390)+(1.0×430)+(1.0×470)+(1.0×510)+0.1**

**o11=390+430+470+510+0.1**

**o11=1800.1**

**Step 6: Compute Sigmoid Activation for o9,o10,o11**

**σ(1720.1) = 1/1+e−1720.1​ ≈ 1.0**

**σ(1760.1) = 1/1+e−1760.1​ ≈ 1.0**

**σ(1800.1) = 1/1+e−1800.1​ ≈ 1.0**

**Thus,**

**o9​=1.0, o10​=1.0, o11​=1.0​**

**Step 7: Compute Output Layer (o12​)**

**The output neuron receives input from o9,o10,o11​.**

**o12​=(o9​×w50​)+(o10​×w51​)+(o11​×w52​)+b**

**o12=(1.0×500)+(1.0×510)+(1.0×520)+0.1**

**o12=500+510+520+0.1**

**o12=1530.1**

**Step 8: Compute Sigmoid Activation for o12 ​**

**σ(1530.1) = 1/1+e−1530.1 ​≈ 1.0**

**Thus,**

**o12​=1.0**

**Final Answer (Forward Propagation Complete)**

| **Neuron** | **Output Before Sigmoid** | **Output After Sigmoid** |
| --- | --- | --- |
| **o1​** | **17500.1** | **1.0** |
| **o2​** | **19000.1** | **1.0** |
| **o3​** | **20700.1** | **1.0** |
| **o4​** | **22400.1** | **1.0** |
| **o5​** | **21840056.1** | **1.0** |
| **o6​** | **22680083.1** | **1.0** |
| **o7​** | **23520106.1** | **1.0** |
| **o8​** | **24360120.1** | **1.0** |
| **o9​** | **1720.1** | **1.0** |
| **o10 ​** | **1760.1** | **1.0** |
| **o11 ​** | **1800.1** | **1.0** |
| **o12​** | **1530.1** | **1.0** |
|  |  |  |

**Final Observations:**

* **Every neuron outputs 1.0 after the sigmoid activation because all pre-activation values are very large, causing the sigmoid function to saturate at 1.0.**
* **This is a classic vanishing gradient issue, meaning weight updates during backpropagation will be near zero.**
* **Solution: Instead of sigmoid, ReLU activation should be used for hidden layers to avoid saturation.**

**​**