

# 機械学習入門

---

Neural Network/Deep Learning



# Deep Learningとは

---

まずはDeep Learningについて理解するためにその歴史から紐解いていくことにしよう。





# History Of Deep learning

# DeepLeaningの変わりゆく名前

---

深層学習。その始まりは1940年代まで遡る。

実は深層学習と呼ばれるようになったのは比較的最近のこと。それ以前は、サイバネティクス(1940年代~1960年代)やコネクショニズム(1980年代~1990年代)と呼ばれ黎明と終焉を繰り返してきた。

今日知られているディープラーニングはそれらの流れを汲み、2006年に再燃した。

さらに細かくみていこう。



# DeepLeaningの誕生初期

---

サイバネティクス期(1940年代~1960年代)では、人工ニューラルネットワークの研究が盛んだった。人工ニューラルネットワークとは人間の脳を神経科学の観点から解析し、モデル化したもので、それを用いることで機械が人間のように分類・回帰を行えるのではないか？人間の脳を解析できるのではないかと期待された。

その頃に考え出されたモデルが、McCulloch-Pittsモデル(McCulloch and Pitts, 1943)や単純パーセプトロンと呼ばれる単層の線形ニューラルネットワークモデルであった。

# DeepLeaningの誕生初期

---

しかし、これらのモデルには欠点があった。SVMで見たように線形モデルではXORのような非線形分離面をもつデータの識別が行えない、という点である。これを指摘したのが(Minsky, and Papert, 1969)の論文で、これによりニューラルネットワークの人気の大きく落ち込むことになった。

とはいえこの当時生まれた確率的勾配降下法(勾配降下法のもっと速いやつとってくれて良いです)のような、今日にも使われる強力な学習アルゴリズムが誕生したのは特筆すべきことであろう。



# DeepLeaningの誕生中期

---

線形ニューラルネットワークモデルが下火となり、約20年後コネクショニズムという波が来る。これは認知科学の文脈で発達し、そこで単純だったニューラルネットワークをさらに複雑にするアイデアが生まれた(多層・多結合)。複雑になる分学習が大変になると直感的に感じると思うが、この当時誕生した誤差逆伝播法をうまく用いることで計算を行なっていた。(これまでの話だと、特徴空間を考えて表現力を上げるということに似ている)

とはいうものの、やはり計算負荷や経済的困難からこの波も下火になっていく。

# 現代のDeepLeaning

---

そこで、GPUやCPU, RAMなどのハードウェアの発達やAdamのようなアルゴリズムの改良、ビッグデータの登場などによって2006年再び深層学習は息を吹き返した。

今日用いられる深層学習という用語は、コネクショニズム期やサイバネティクス期では人間の脳を模倣したニューラルネットワークがベースとなっていたが、現代では多様化し単に層が深い構成の学習アルゴリズムを指している。



# 忙しい人のためのDeepLeaningの歴史まとめ

---

(1940-1960)

単純な人工ニューラルネットワーク(ANNs)が生まれる。(モデルとして弱い)

(1980-1990)

それが複雑になる。(計算できない、金もない)

(2006~)

ニューラルネットワーク模倣に限らず、色々な文脈で使われる・使えるようになる。

# Neural Networkとは

---

まず、深層学習の基礎となる人工NeuralNetwork(ANN)についてみていこう。

そのために、人間が何かを識別する、ということに関して話していく。



# Neural Networkとは

---

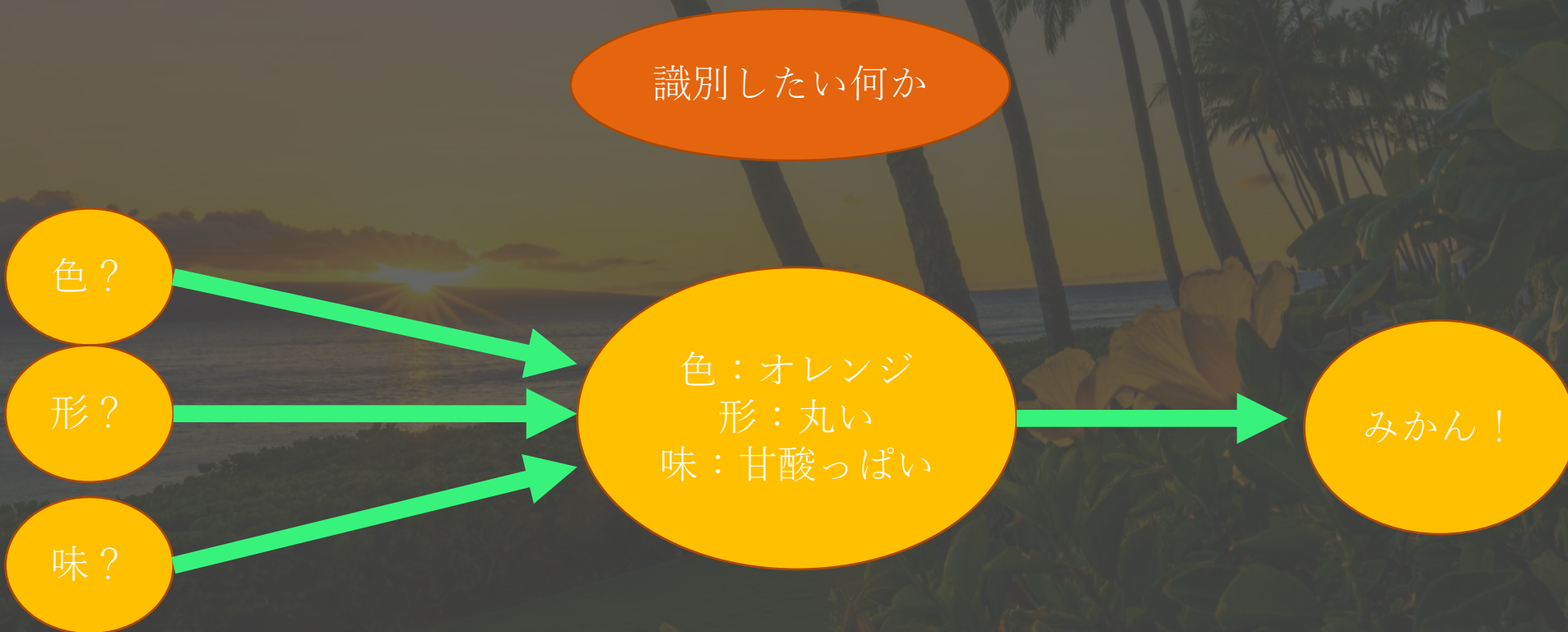
情報が不足していて識別できない



# Neural Networkとは

---

十分な情報が与えられて、識別できた！





# Neural Networkとは

---

つまり先の例を考えると、いくつかの情報が組み合わさって、その情報量がある値(閾値)を超えると識別できた、ということ。

それを式にすると、

$$y = \sum \text{情報},$$

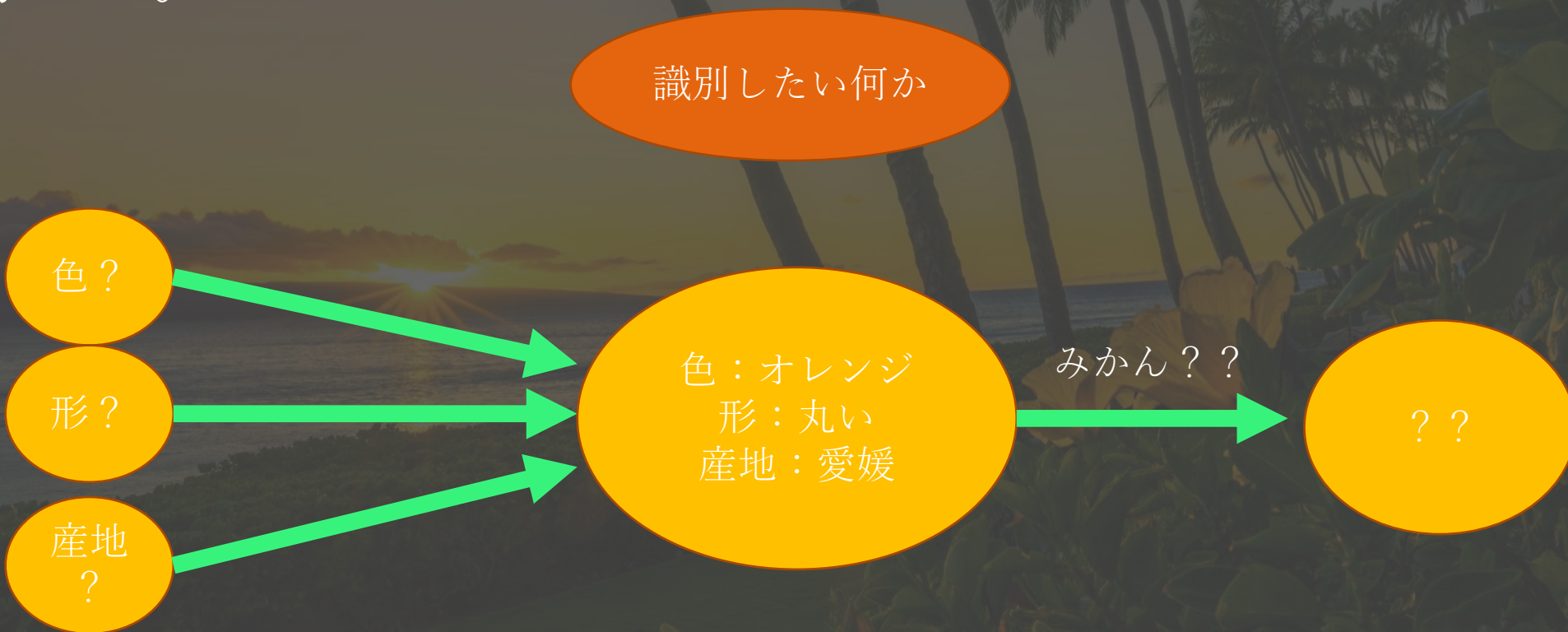
*if*  $y \geq \text{閾値}$ : みかん  
*else*: みかんでない

という形で書けるだろう。



# Neural Networkとは

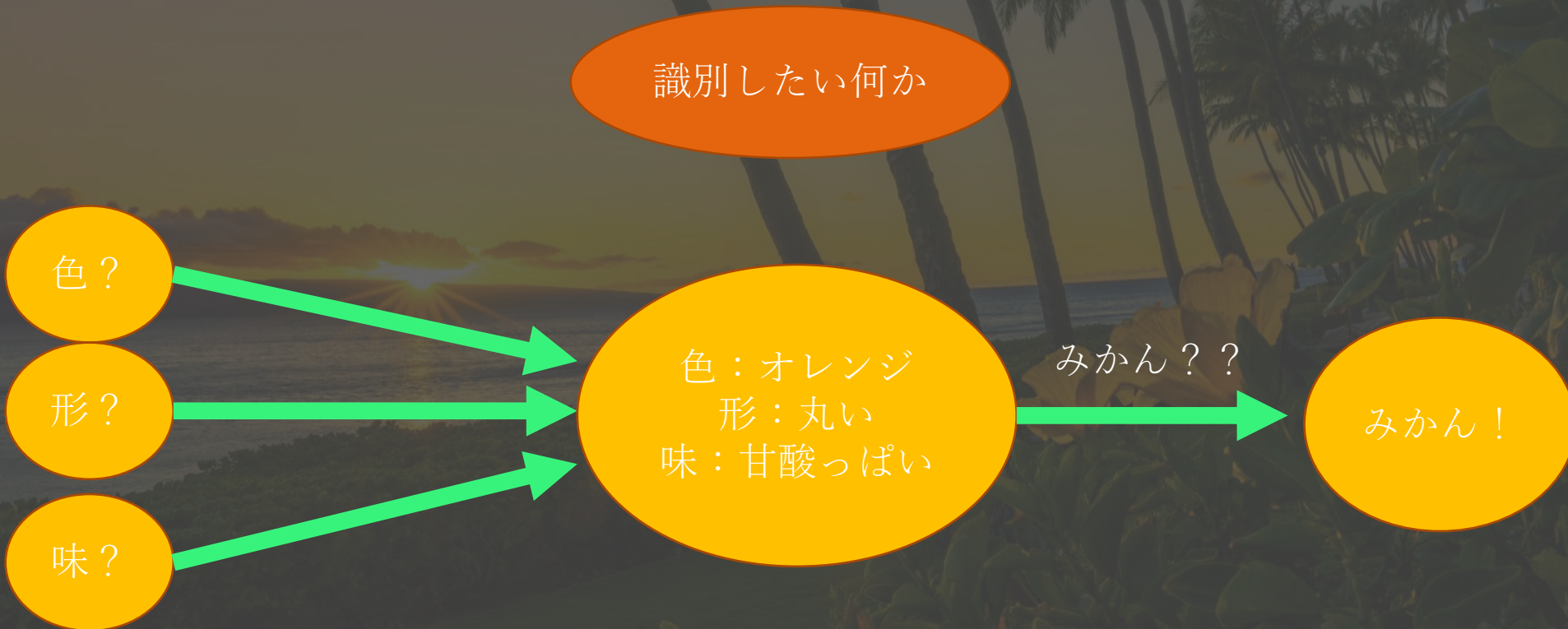
さらに少し考える。情報は全て等価だろうか？例えば下の例を考える。





# Neural Networkとは

味と産地だったらみかんということを識別する上で圧倒的に味ではないだろうか？





# Neural Networkとは

---

何が言いたいのかというと、下の式のように情報によって重みが違うんじゃないか、ということ。

$$y = \sum \text{重み} \times \text{情報},$$

*if*  $y \geq$  閾値: みかん

*else*: みかんでない

よって最終的に次のように書けるだろう。

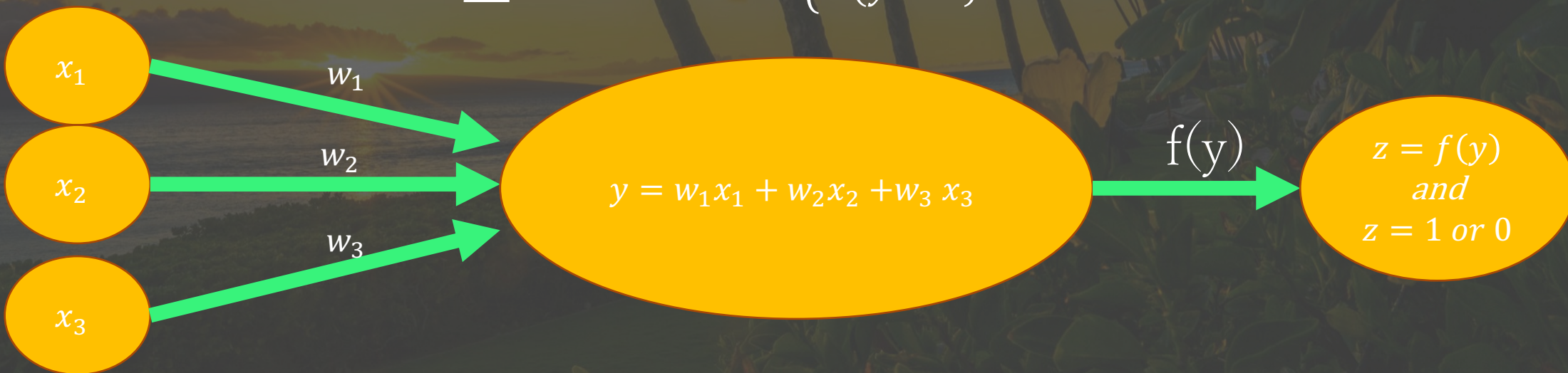


# Neural Networkとは

何が言いたいのかというと、下の式のように情報によって重みが違うんじゃないか、ということ。

そして、与えられた情報を識別のための関数(活性化関数)があるんじゃないか、ということ。

$$y = \sum w x, \quad f(y) = \begin{cases} 1 & (y \geq 0) \\ 0 & (y < 0) \end{cases}$$





# Neural Networkとは

---

このような構造でモデル化(**形式ニューロン**としてモデル化)できるだろうと考えたのがMcCullochとPittsである。

MPモデルでは重みを人手で設定していたが、重みの学習に勾配降下法のような最適化アルゴリズムを用いたものがパーセプトロン(Rosenblatt, 1958, 1962)と呼ばれるモデルである。

ちなみに、、、ロジスティック回帰は閾値がある確率となるパーセプトロンである。



# Neural Networkとは

---

パーセプトロンとして重要な性質は、

1. 重み付きのデータの足し算で計算される。
  2. 計算された重み付き和が**活性化関数**によって識別される。
- という点である。

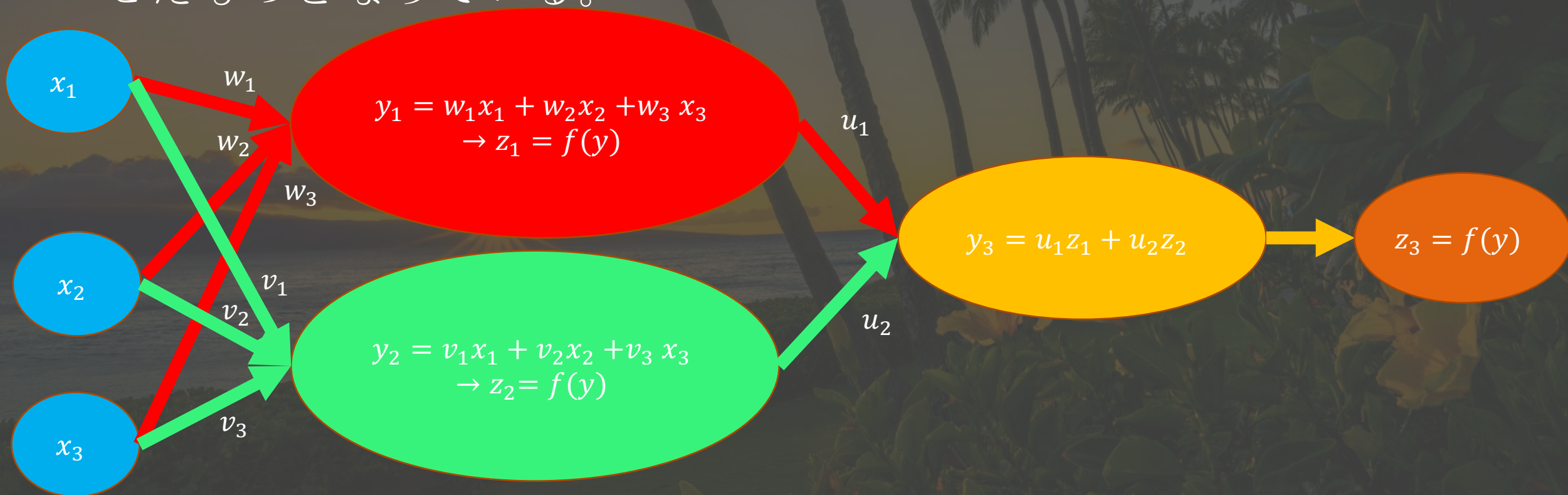
特に前のスライドのように簡単な構造のパーセプトロンを**単純パーセプトロン**という。

そして、次のコネクショニズム期に考え出されたものが次のスライドで示すような構造のパーセプトロンである。



# Neural Networkとは

複雑なパーセプトロンの例：単純パーセプトロンを組み合わせたものとなっている。





# Neural Networkとは

---

複雑なパーセプトロン(**多層パーセプトロン(MLP)**という)の特徴は、

1. 重み付きのデータの足し算で計算される。
2. 計算された重み付き和が**活性化関数**によって識別される。

という2点に加え、

3. **単純パーセプトロン**がいくつも組み合わさった構造を持つ。

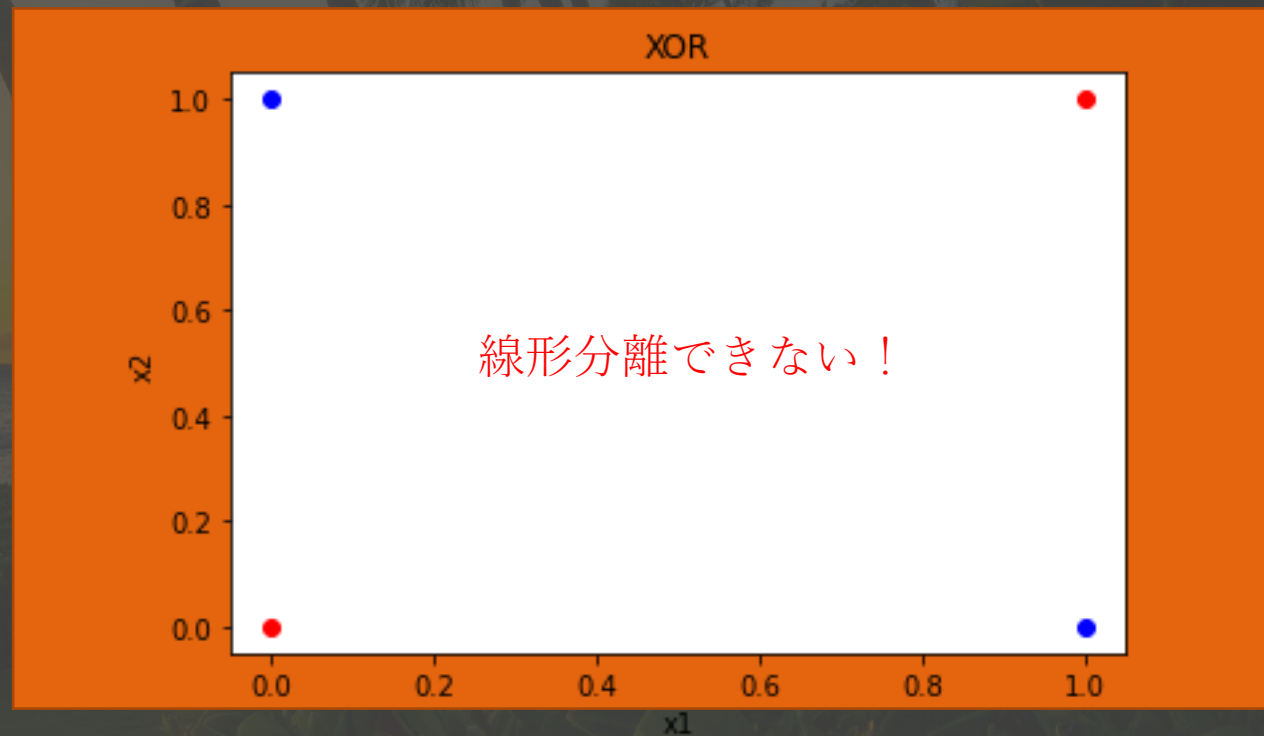
というものがある。

DL史でみたようにこれによって線形識別だけでなく非線形な識別も可能になっている。次は簡単なデータでそれをみていこう。

# Neural Networkとは

## XORデータの識別

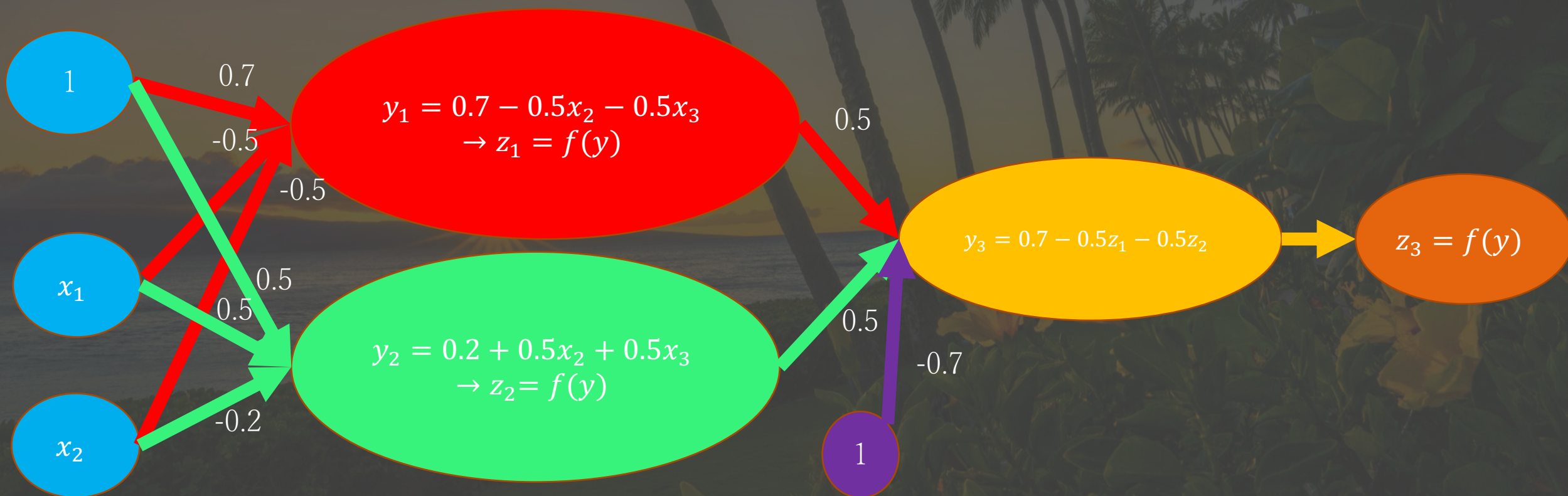
$x_1$	$x_2$	$y$
1	1	0
1	0	1
0	1	1
0	0	0





# Neural Networkとは

次のようなMLPで線形分離できることを確認しよう。





# Neural Network (subject1)

---

- XORのデータがMLPで識別できることを確認しよう。

例えば、 $x_1 = 0, x_2 = 0$ の時

$$y_1 = 0.7 - 0 - 0 \rightarrow z_1 = 1$$

$$y_2 = 0.2 + 0 + 0 \rightarrow z_2 = 1$$

$$y_3 = 0.7 - 0.5 - 0.5 = -0.3 \\ \rightarrow z_3 = 0$$

という形で(1, 1)->0と識別できている。





# Neural Networkとは

---

先の例では、MLPを最適化した結果の重みを使ってXORデータが識別可能であることを確認した。

このように、MLP(NN)の重み最適化していくものがいわゆる **深層学習** である。

\* \* \*

補足：MLPとNNの違いは、厳密ではないが、活性化関数に後に示すSTEP関数のみを用いているかどうかということと、全結合型であるかどうか、という認識をしている。(原著でそうだったから)



# Neural Networkとは

---

活性化関数の例:

多くは閾値に応じて「ガクッ」と変わるような性質を持つ。

一般にどれがいいかは経験的に決める。(物体認識ではAbsolute value rectificationが使われたりもするが、、、)

<https://colab.research.google.com/drive/1IdYYWiWIekV5wU9yGZAomjTmmcjHO8u8?usp=sharing>



# Neural Networkとは

---

一般的なMLP(NN)の構造



# Neural Networkとは

---

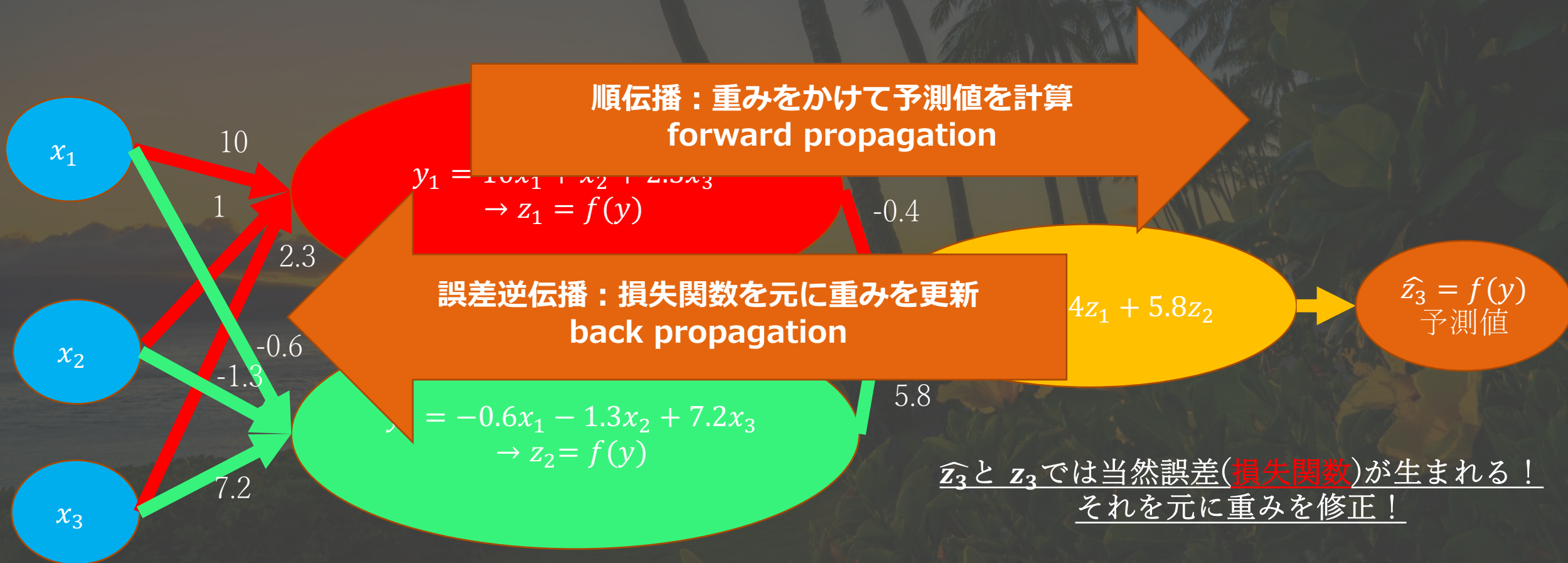
最終的に僕らが何をしたいのかというと、先のようなMLP(NN)構造の中の重みを最適化していきたい。

XORの例だと、 $y$ という教師データに沿って、重みをいじりながら変えて、正しく識別できるようにしていた。

では、どのように最適化しているのかを詳しくみていこう。



# Neural Networkとは



# Neural Networkとは

---

Pytorchを用いた実装

1. 中間層と出力層を定義
2. 順伝播を定義
3. 最適化の回数(epoch数)を定義
4. 勾配を初期化
5. 順伝播
6. 誤差逆伝播
7. 重みを更新
8. 4-7をepoch数繰り返す



# Neural Network (subject2)

---

XORが先のNNで識別できることを確認しよう。





# Neural Network (subject3)

---

cancerデータセットを用いて、  
pytorchによる  
LogisticRegressionモデルを作  
成

終わった人は中間層を追加する  
などしていじって遊んでみよ。





# Neural Networkとは

---

まとめ

脳の神経構造を模した機械学習モデル

順伝播によって生じた誤差を逆伝播 & 勾配の更新、重みの更新によって解消していく