Classifiers

$Q(s,a)$ = expected reward given state s taking action a

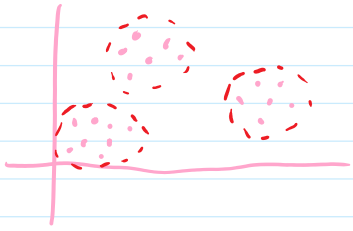
Regression: function attributes → value

Classifier: function attributes → class
- class: species of iris (virginica, setosa, versicolor)
- class: digit (0 ... 9)

Iris flower data set

MNIST database http://yann.lecun.com/exdb/mnist/

Learning:
- supervised – examples; generalize from these examples
  - ↳ grandmaster moves
- reinforcement – observe state transitions, rewards earned
- unsupervised – discover unknown relationships

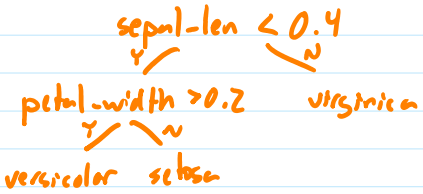


Methods:

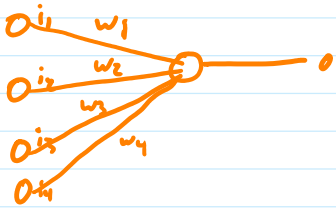
k-nearest neighbors

2 of nearest 3 are ● so predict ●

decision trees
nested ifs



perceptron



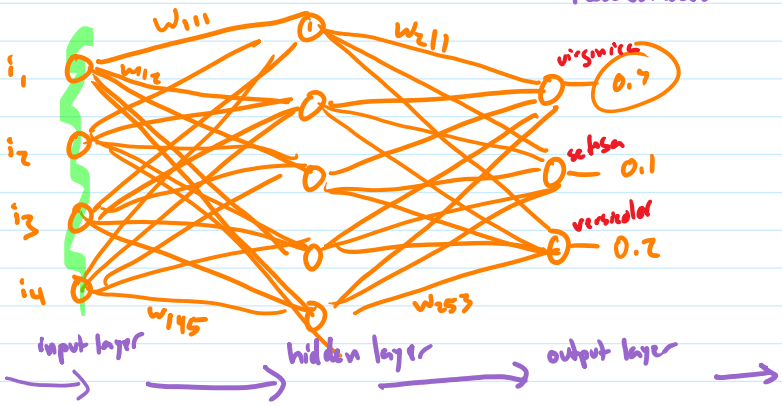
$$\text{output} = f\left(\sum_{j=1}^{k} i_j \cdot w_j\right)$$

↳ $f(x) = \begin{cases} 1 & \text{if } x \geq b \\ 0 & \text{otherwise} \end{cases}$

multi-layer perceptron (artificial neural network)
feed forward

multi-layer perceptron (artificial neural network)

feed forward



Input / Output

categorical

versicolor = 0
virginica = 1
setosa = 2
yalrca =

versicolor!
0

50/50 versicolor/virginica	75/25 versicolor/virginica	50/50 virginica/setosa	50/50 versicolor/setosa
0.5	0.25	1.5	2.5

one-hot — 1 input/output per class

	versicolor	virginica	setosa			
←	1	0	0	0.5	.75	output = prob that input belongs to class
←	0	1	0	0.5	.25	
←	0	0	1	0	0	

date

seconds since epoch (normalized) → arbitrarily chosen starting point (Jan 1 1970)

3" Jan 1 1900 = −2208988800
1" Jan 1 1999 = 915148800 → normalize: $\frac{-2.2 \text{ billion}}{4 \text{ billion}}$ $\frac{915 \text{ million}}{4 \text{ billion}}$

predict snowfall given date

month / day / year

Jan 1 1900 = 1 1 1900
Jan 1 1999 = 1 1 1999 — very different
Jan 31 2020 = 1 31 2020
Feb 1 2020 = 2 1 2020
Dec 31 1998 = 12 31 1998

fuzzy

	Jan	Feb	Mar	...								Dec
	0	0	0	0	0	0	0	0	0	0	0	0
Jan 1	.52											.48
Jan 10	.7											.3
Mar 15			1.0									
Dec 31	.48											.52

rolls 1 1 3 4 6
1 1 1 2 5

ANN Supervised Learning

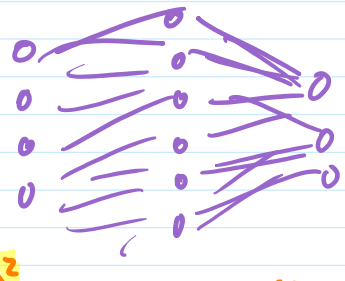
initialize weights randomly

until trained
  for each example in training data
    compute outputs (send input through ANN, get output)
    compute total error (difference between output and correct results)
    adjust weights to reduce error (calculus)

current predictions

regression

classification



output	correct
.7	1
.25	0
.05	0

mean squared error (MSE) $\frac{1}{n}\sum_{i=1}^{n}(Correct_i - Predict_i)^2$

cross-entropy

Actual		Predicted	MSE	CE		Predicted	MSE	CE
0		0.25	0.0625	0		0.001	0.000001	0
0		0.25	0.0625	0		0.001	0.000001	0
1		0.25	0.5625	-0.60206		0.3799	0.384524	-0.42033
0		0.25	0.0625	0		0.6199	0.384276	0
			0.1875	0.60205999			0.192201	0.420331

MSE prefers prediction 1

CE prefers prediction 2