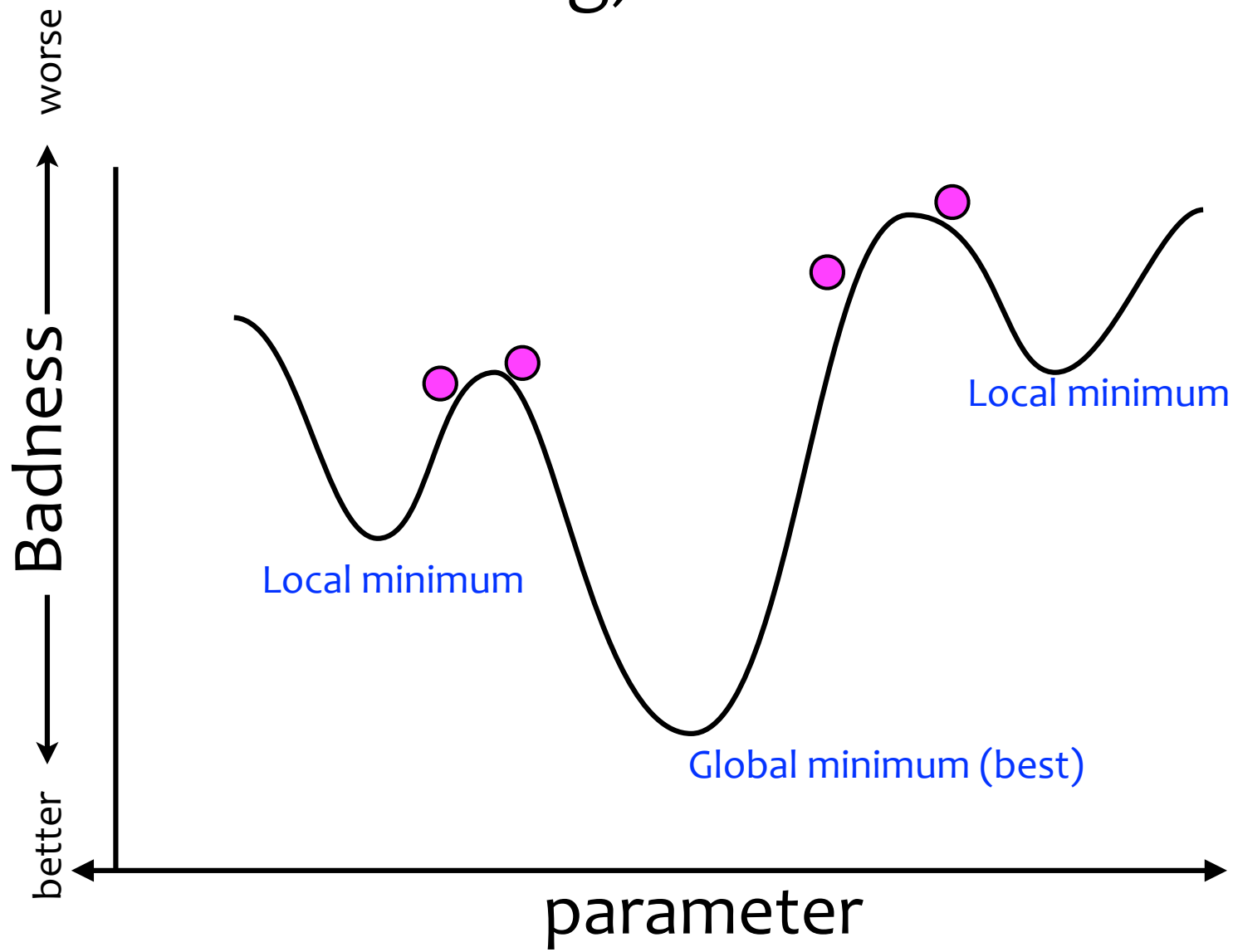
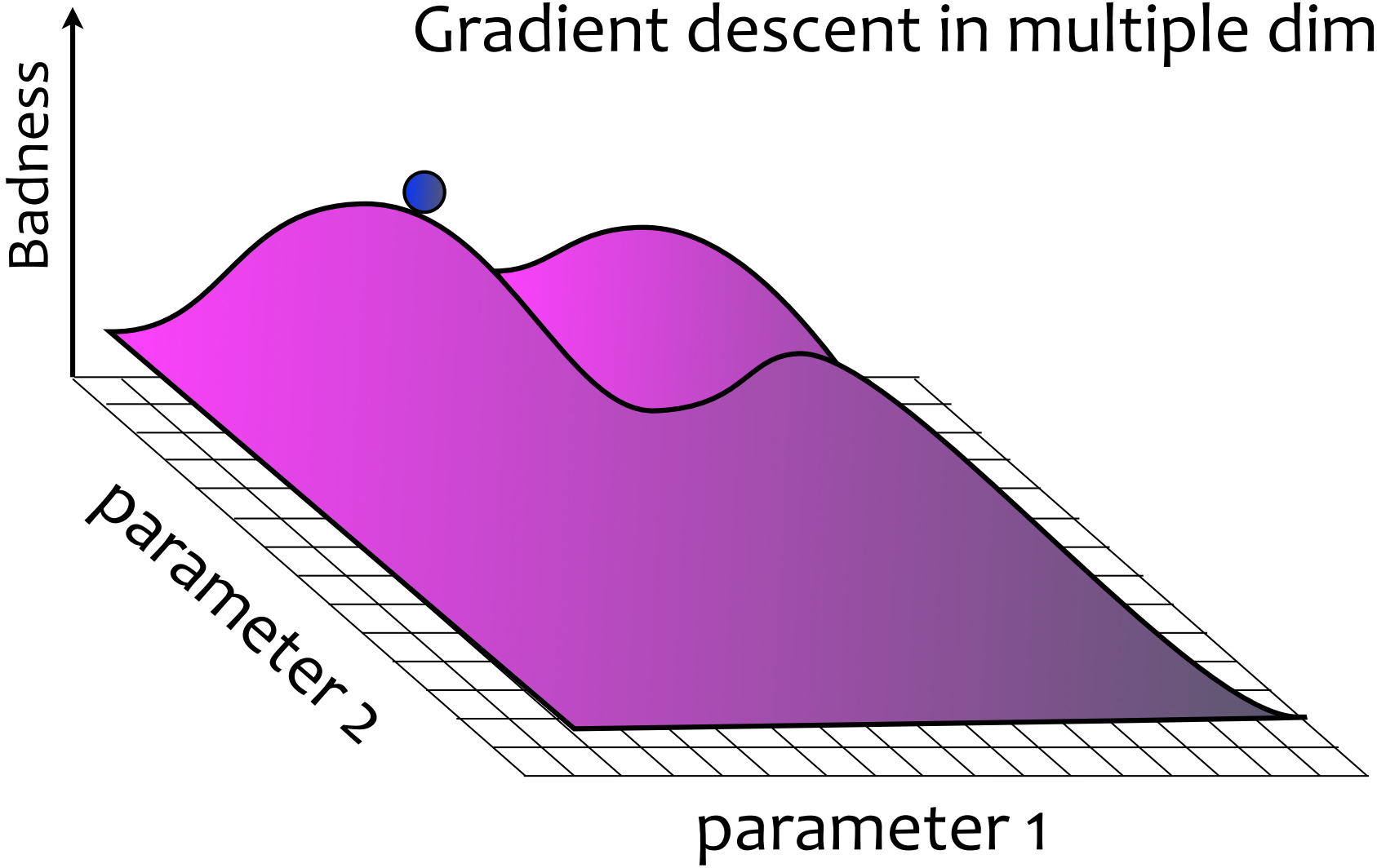
Hill Climbing, aka Gradient Descent



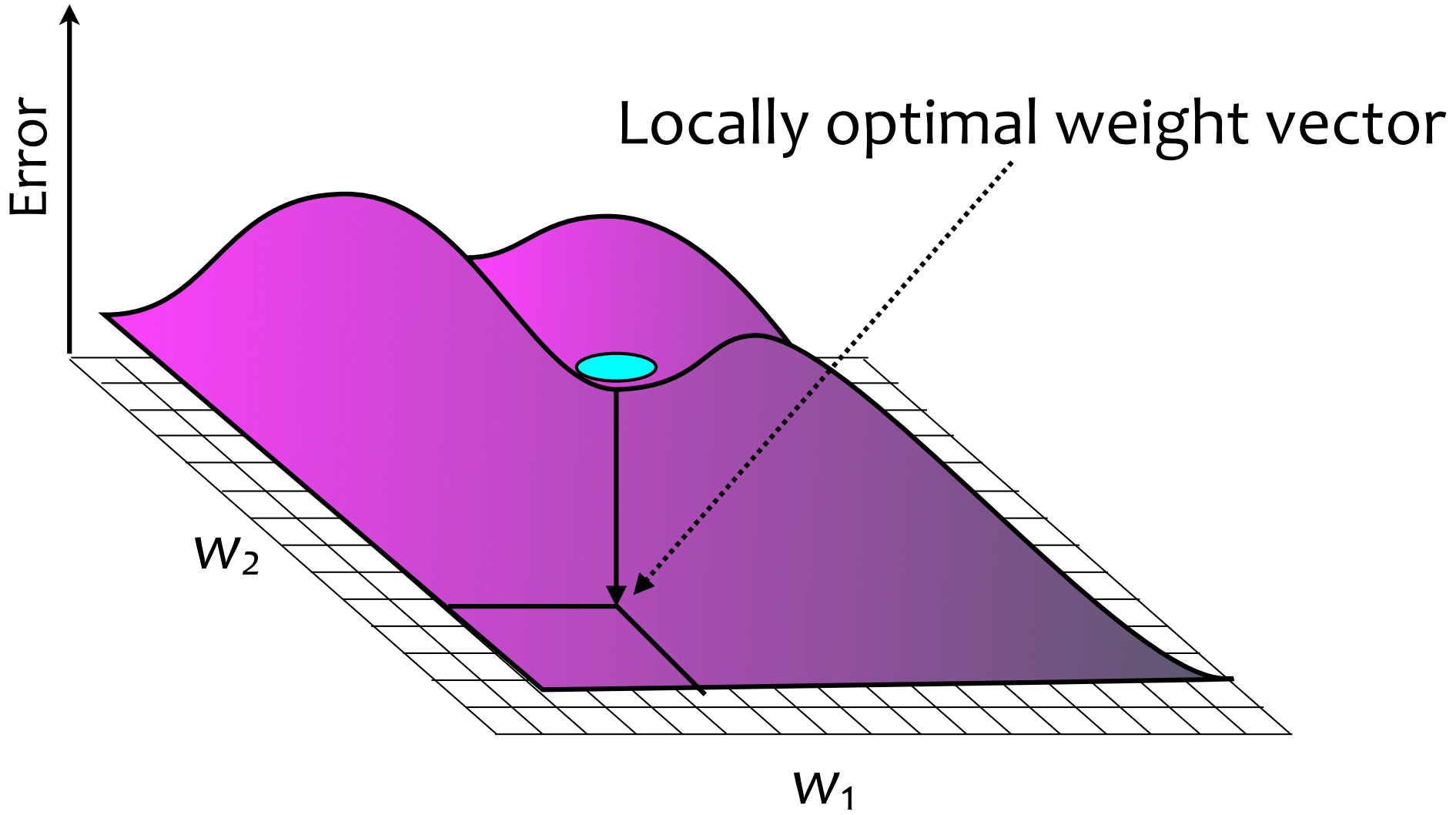
Gradient descent in multiple dimensions



Each location on the grid is one parameter combination that must be considered.

There are K^N of them (K positions along each of N dimensions)

Neural network learning as search through weight-space



The backlash

- In the late 1980s and 1990s, there was a ferocious backlash against connectionist models
- Some argued that connectionist models:

  - were not remotely biologically plausible

  - Could not explain real human data

  - Could not even in principle explain classic phenomena, including:

    - symbolic representation of concepts

    - compositionality

    - one-shot learning

- Connectionism went out of style for a while

PDP models: pro

- PDP models are biologically plausible.

  - The brain actually is a network of nodes organized in layers.

- Graceful degradation

  - Small modifications of the network correspond to small differences in function

- The approach is parsimonious

  - a few simple rules explain learning across all domains

  - Rules, symbols, are epiphenomenal .

PDP models: con

- PDP models aren't actually biologically plausible

  - E.g., back-propagation of error doesn't seem to have a neural correlate

  - They model a gradual learning process which isn't how people learn

  - They model learning at the wrong level of explanation (cf. Marr)

- PDP learning is just gradient descent. Not guaranteed to work, and takes a long time. People don't take a long time.

- PDP models overfit. With enough hidden nodes, any function can be simulated.

- PDP models have no principles; they model a complex system by another complex system;

  "The best material model for a cat is another cat, or preferably the same cat." - Arturo Rosenblueth

  - Like a full-sized map of the world - Jorge Luis Borges

- PDP models can't handle compositionality

  - Finite number of response categories. No symbols.

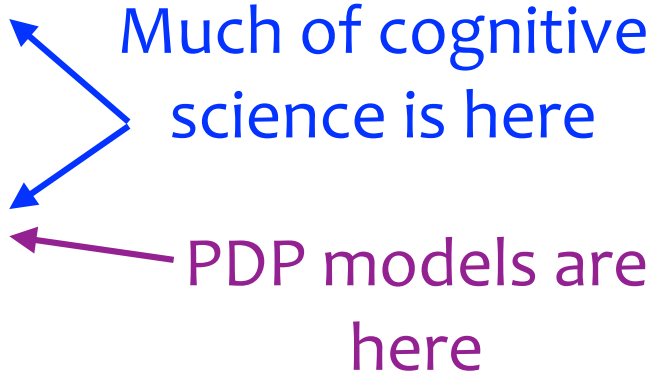
Marr's levels of explanation

- David Marr (an influential vision theorist) suggested that computational systems can be understood on three different levels:

  - Theory of the computation: what problem is the system solving? What assumptions does it make to solve it?

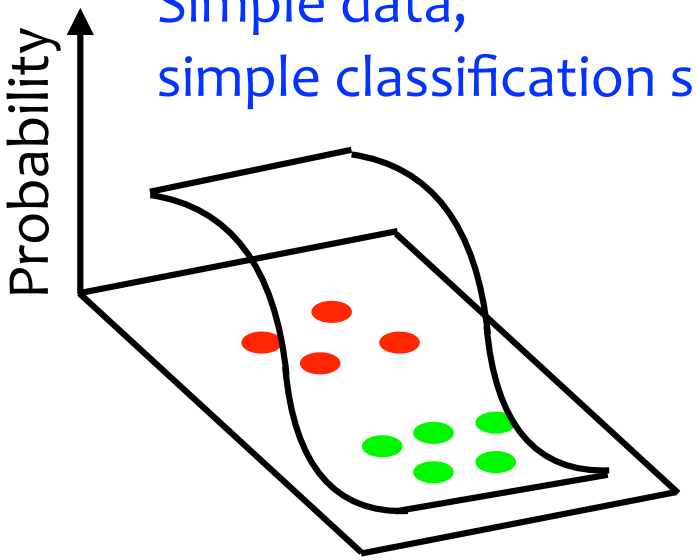
  - Algorithm: what sequence of computational steps does it use to solve the problem?

  - Implementation: How is the algorithm physically implemented?

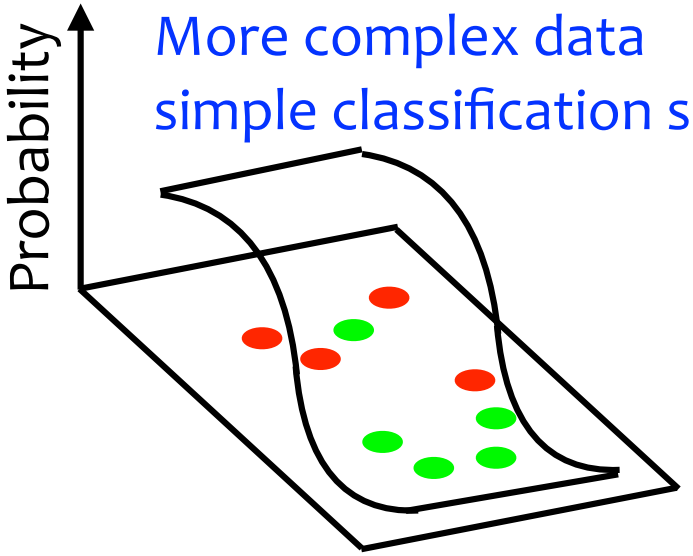


Overfitting in NN learning

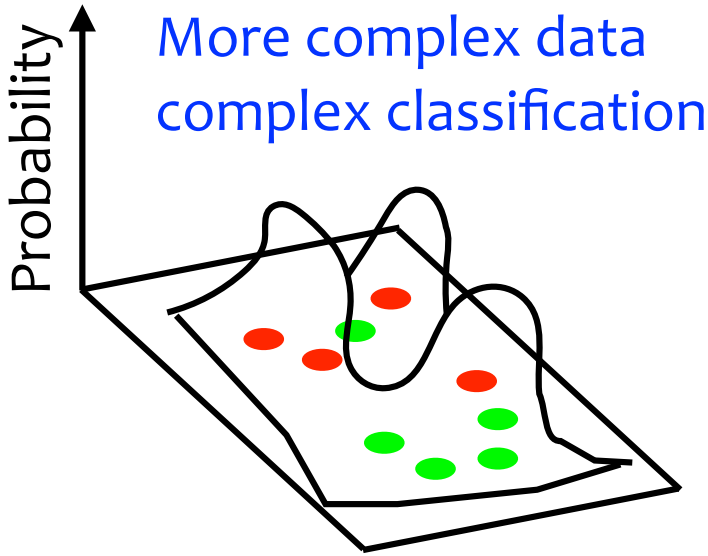
Simple data,
simple classification surface



More complex data
simple classification surface



More complex data
complex classification surface?



- More complex decision surfaces can overfit the data
- The complexity of the decision surface is controlled by the number of weights in the network, i.e. the number of hidden nodes