Outline

- Problems with neural networks
- Support Vector Machines
- Controlling complexity in statistical models

Questions about neural networks

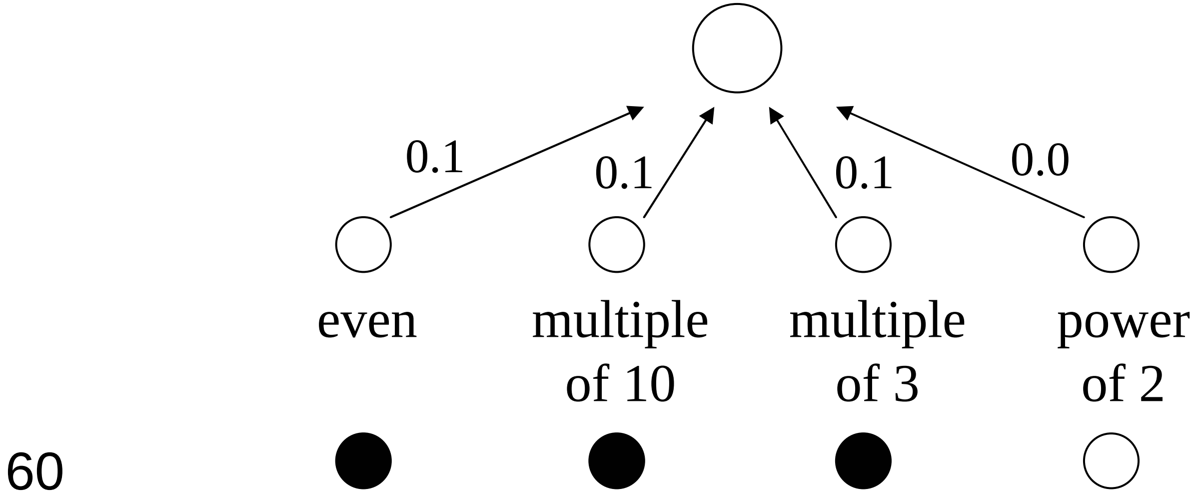
- Why do they have such a bad rap?
- To what extent are neural networks brain-like?
- They take a long time to train. Is that a good thing or a bad thing from the standpoint of cognitive modeling?

Models versus Data

Images removed due to
copyright considerations.

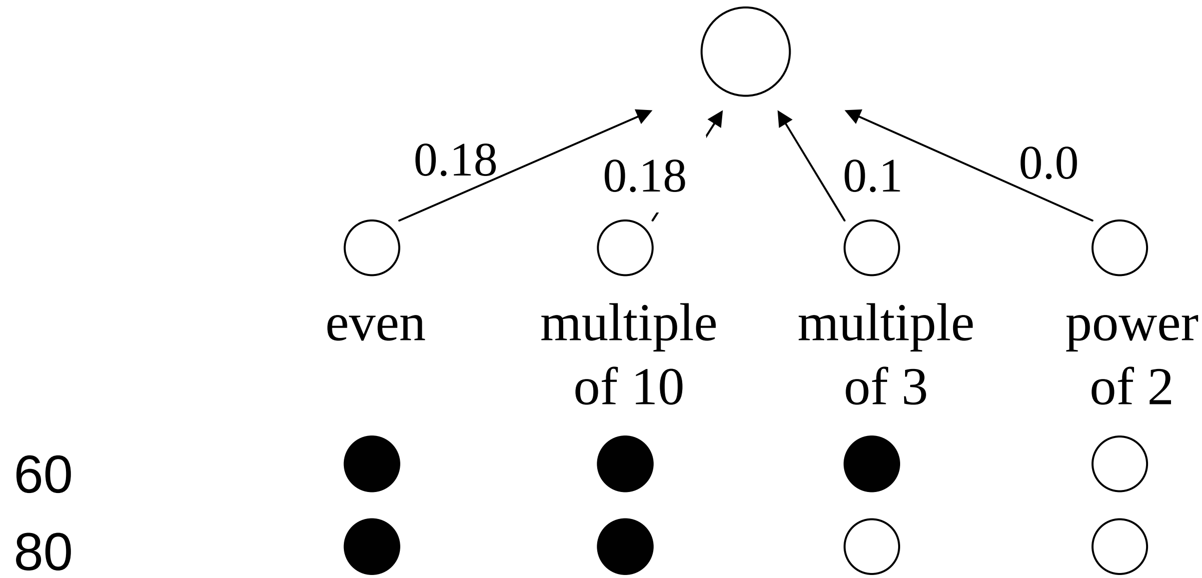
Number game

- Neural networks
  - Delta rule learning



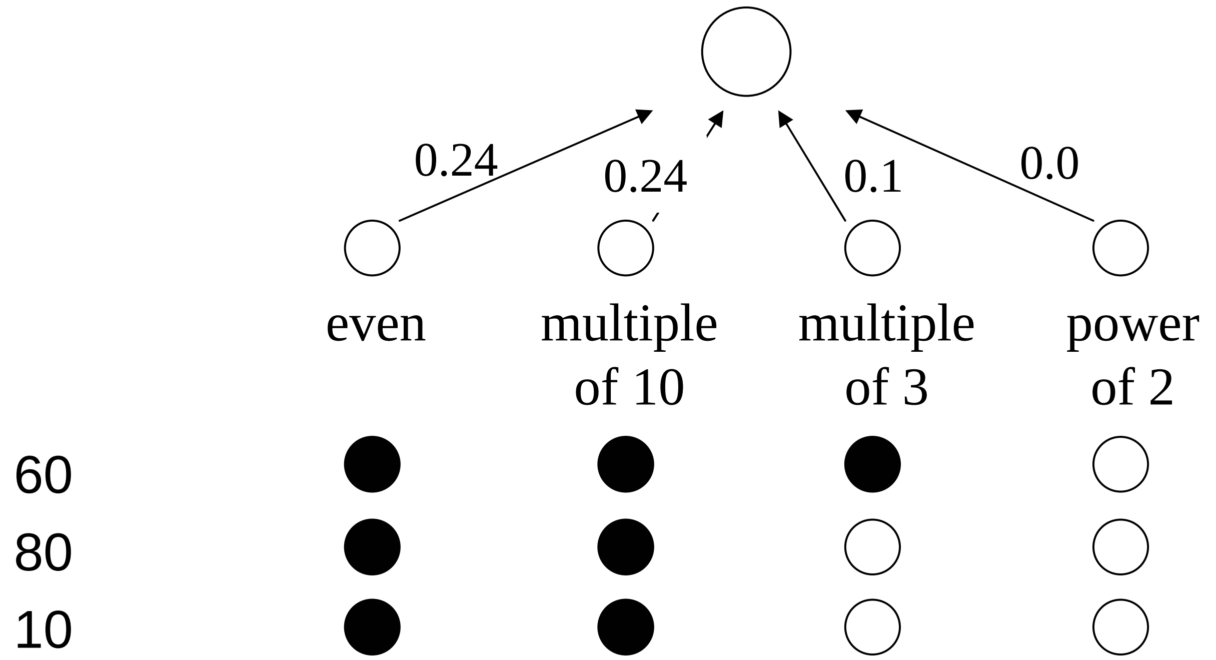
Number game

- Neural networks
  - Delta rule learning



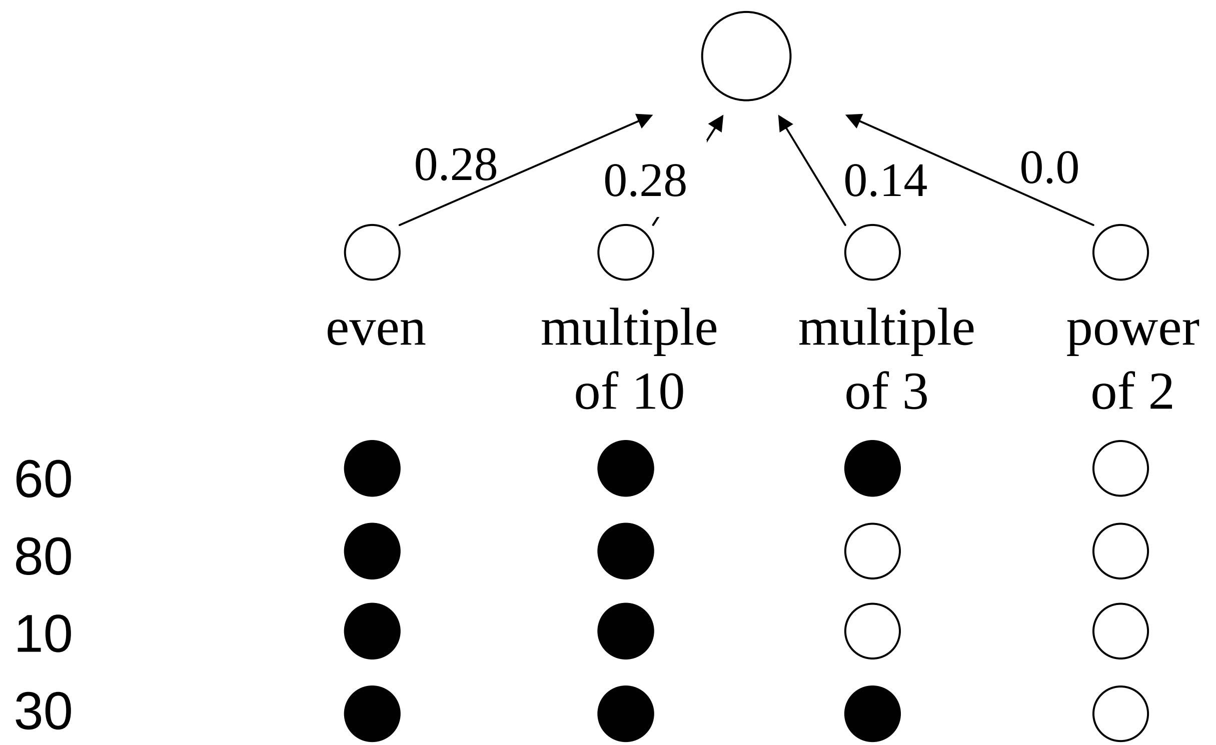
Number game

- Neural networks
  - Delta rule learning



Number game

- Neural networks
  - Delta rule learning



Alternative models

- Similarity to exemplars
  - Average similarity: $p(y \in C \mid X) = \frac{1}{|X|} \sum_{x_j \in X} \mathrm{sim}(y, x_j)$

60

60 80 10 30

60 52 57 55

Images removed due to copyright considerations.

Data

Model (r = 0.80)

Images removed due to copyright considerations.

Bayes (with basic-level bias)

Bayes (*without* basic-level bias)

Images removed due to copyright considerations.

Questions about neural networks

- Why do they have such a bad rap?
- To what extent are neural networks brain-like?
- They take a long time to train. Is that a good thing or a bad thing from the standpoint of cognitive modeling?

Image removed due to copyright considerations.

(Kruschke)

Outline

- Problems with neural networks
- Support Vector Machines
- Controlling complexity in statistical models

Support Vector Machines (SVMs)

- Problems with neural networks
  - Flexible nonparametric classifiers, but slow to train and no good generalization guarantees
- Problems with perceptrons
  - Good generalization guarantees and fast training, but only for a limited parametric family of problems (linearly separable classes).
- SVMs seek the best of both worlds.

The virtue of high-dimensional feature spaces

Image removed due to copyright considerations.

The virtue of high-dimensional feature spaces

Image removed due to
copyright considerations.

The SVM approach

- Embed data in *d*-dimensional feature space ($d >> \#$ data points, maybe infinite).
- Find optimal separating hyperplane in feature space.
- What makes this possible:
  - For *d* large enough, all categorization problems become linearly separable.

The SVM approach

- Embed data in *d*-dimensional feature space ($d >> \#$ data points, maybe infinite).
- Find optimal separating hyperplane in feature space.
- What makes this possible:
  - Computations depend only inner products between feature vectors, which can be expressed as a simple kernel on inputs, e.g.:

$$\mathbf{z}^{(i)} \cdot \mathbf{z}^{(j)} = (\mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)})^2$$

The SVM approach

- Embed data in *d*-dimensional feature space ($d >> \#$ data points, maybe infinite).
- Find optimal separating hyperplane in feature space.
- What makes this possible:
  - A wide range of simple kernels define very high-dimensional (and useful) feature spaces:

$$\mathbf{z}^{(i)} \cdot \mathbf{z}^{(j)} = (1 + \mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)})^k$$

$$\mathbf{z}^{(i)} \cdot \mathbf{z}^{(j)} = \exp(- \| \mathbf{x}^{(i)} - \mathbf{x}^{(j)} \|)^2$$

The original Perceptron idea

- Embed data in *d*-dimensional feature space ($d >> \#$ data points, maybe infinite).
- Find optimal separating hyperplane in feature space.
- Problems:
  - Didn’t know the “kernel trick”, but inspired by neural receptive fields. (c.f. Minsky & Papert)
  - Didn’t have a good concept of “optimal separating hyperplane”. *In high-dimensional feature spaces, infinitely many errorless planes.*

Maximum margin hyperplane

- Depends only on the “support vectors”: points closest to the boundary between classes.
- PAC-style guarantees of good generalization:
  - log |H| ~ # of support vectors

Image removed due to copyright considerations.

SVMs and neural networks

- SVMs have many of the attractive features of neural networks, but not all.
  - No sharing of weights (parameters) across many related learning tasks.

Image removed due to copyright considerations.

SVMs and neural networks

- SVMs also preserve some of the limitations of neural networks.
  - No learning from just one or a few positive examples.
  - No natural way to build in prior knowledge about categories.
  - No explicit representation of learned concepts or abstractions.

Evaluating models for concept learning

- Dimensions:
  - Causal versus Referential inference
  - Parametric versus Non-parametric
  - Generative versus Discriminative
- Which of these approaches are most suited for understanding human learning?

- Dimensions:
  - Causal versus Referential inference
  - Parametric versus Non-parametric
  - Generative versus Discriminative
- Issues:
  - All-or-none versus graded generalization
  - Learning from very few labeled examples
  - Incorporating unlabeled examples
  - Incorporating prior knowledge
  - Forming abstractions and theories
  - Learning “new” concepts
  - Trading off complexity with fit to data

Outline

- Problems with neural networks
- Support Vector Machines
- Controlling complexity in statistical models

Overfitting in neural networks

Image removed due to copyright considerations.

Overfitting is a universal problem

- Concept learning as search: subset principle
- Bayesian concept learning: size principle
- Categorization with generative models

Image removed due to
copyright considerations.

- Categorization with discriminative models

Image removed due to
copyright considerations.

How to control model complexity?

- Traditional “model control parameters”
  - Early stopping
  - Weight decay
  - Slow learning rate
  - Bottleneck number of hidden units

Image removed due to copyright considerations.

How to choose control parameters?

- Cross-validation
  - Separate data into "training set" and "validation set" (simulated test data)
  - Learn on training set until validation error stops decreasing.

Image removed due to copyright considerations.

Cross-validation

- Advantages:
  - Intuitive
  - Works in practice
- Disadvantages
  - Theoretical justification unclear.
  - Unclear how to choose training/validation split.
  - Doesn’t use all of the data.
  - Difficult to apply to many control parameters.

Monte Carlo Cross-validation

- Consider many different random training/test splits.
  - Smythe: Application to choosing the correct number of components in a mixture model.
- Disadvantages
  - Theoretical justification unclear.
  - Unclear how to choose training/validation split.
  - Doesn't use all of the data.
  - Difficult to apply to many control parameters.
  - Slow.