Networks for Approximation and Learning

Tomaso Poggio
Federico Girosi

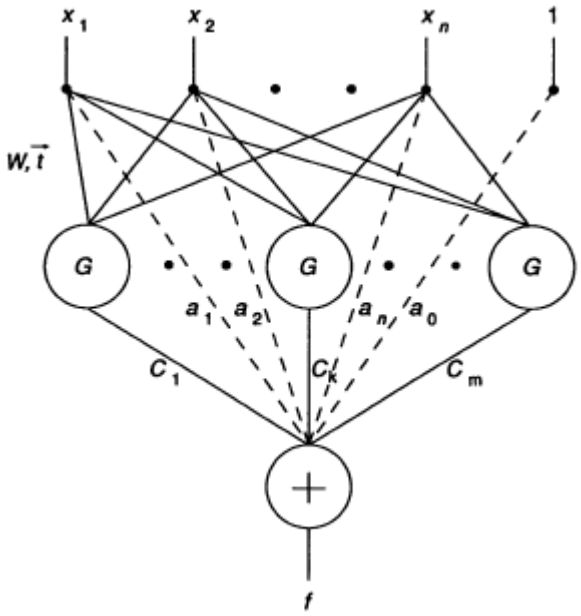
Jin Huang
March 3, 2010

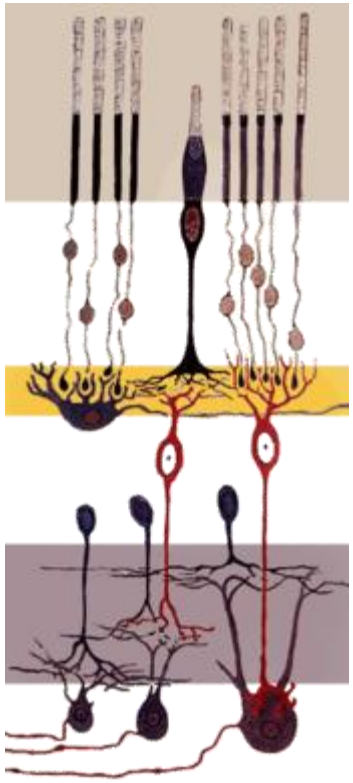
Outline

1. Big Picture
2. Background
3. Why is that
4. Related Work
5. Summary

Big "Picture"

Regularization Algorithm





Establish the connection between the neural network with the classical mathematics number approximation method(Regularization, general splines, etc).

Background

A lot of network had been developed to do things like the input-output mapping, multivariate function approximation and hypersurface reconstruction.

Regularization techniques can help generalize this kind of network algorithm, and may help show the essence of the network and approximation schemes.

All about approximation

Approximation

From f(X) to F(W,X)

X is the inputs,
W is the parameters,
And F is the scheme(function) for approximation.

All about approximation

Approximation

From f(X) to F(W,X)

1. Which F to use;
2. How to find the optimal W;

All about approximation

Approximation

From f(X) to F(W,X)

The network approach(3-layers):

$$F(W, X) = \sigma\left(\sum_n w_n \sigma\left(\sum_j u_j x_j\right)\right)$$

All about approximation

Then to find the solution z to minimize:

$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

How close to samples **plus** Smoothness(assumption)

P is an operator, usually a differential operator.

λ controls the degree of generalization.

All about approximation

Then to find the solution z to minimize:

$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

Probability distribution description(MAP):

$$P_{z/d}(z;\, d) \propto P_z(z)\, P_{d/z}(d;\, z).$$

Given the data d, the distribution of z is correlated with the

P(z) : Priori probability(Smooth priori assumption)

P(d|z): The noise model.

All about approximation

Then to find the solution z to minimize:

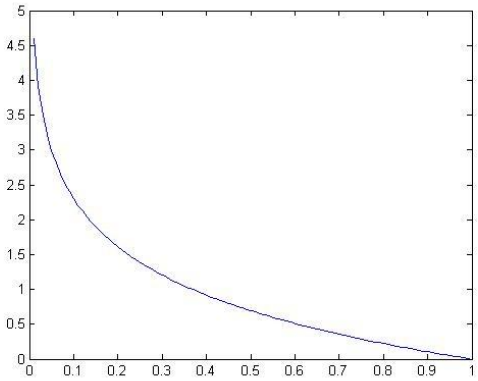
$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

Complexities of hypothesis description:

$$C(z|d) = C(z) + C(d|z) + c$$

Complexity: C(x) = -logP(x)

Try to minimize the complexity.



Solve it: Regularization

Then to find the solution z to minimize:

$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

Euler-Lagrange equations:

$$\hat{P}Pf(x) = \frac{1}{\lambda} \sum_{i=1}^{N} (y_i - f(x))\delta(x - x_i)$$

And Green's function:

$$\hat{P}PG(x; y) = \delta(x - y).$$

Solve it: Regularization

Then to find the solution z to minimize:

$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

Then we solve it:

$$f(\boldsymbol{x}) = \frac{1}{\lambda} \sum_{i=1}^{N} (y_i - f(\boldsymbol{x}_i)) G(\boldsymbol{x}; \boldsymbol{x}_i).$$

Simpler:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\boldsymbol{x}; \boldsymbol{x}_i)$$

Solve it: Regularization

Then to find the solution z to minimize:

$$\sum_i (z_i - d_i)^2 + \lambda \|Pz\|^2$$

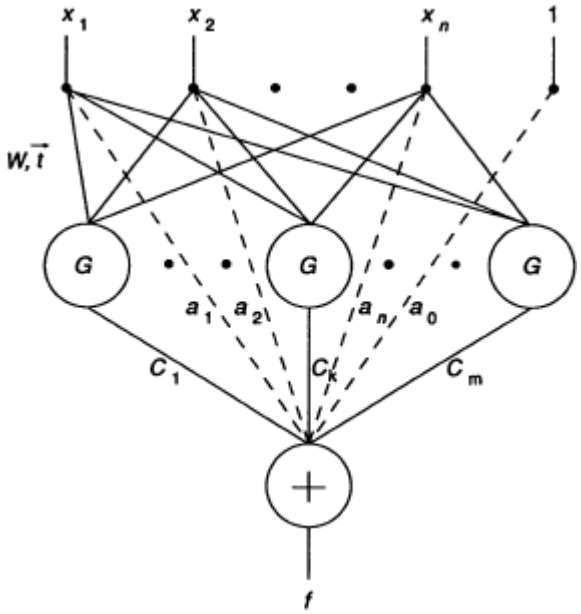
Actually, the operator P is required to be translationally and rotationally invariant, thus G would be **radial** function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$$

Looks familiar? we get connections with RBF here.

Equivalent Network

Remember this?



Nonlinear part:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$$

Extensions:

Still this function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$$

1. The complexity is dependent on the dimensionality of the training set(number of examples), which is very high. So try to make it smaller.
2. Considering the weighted distance.
3. Different functions and scales of G.
4. Moving centers.
5. Learning the negative examples.

Extensions:

Still this function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$$

Move the centers through learning:

$$f^*(\boldsymbol{x}) = \sum_{\alpha=1}^{n} c_\alpha G(\boldsymbol{x}; \boldsymbol{t}_\alpha)$$

Extensions:

Still this function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} c_i G(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$$

The weighted distance(weight of input layer)

$$f^*(\boldsymbol{x}) = \sum_{\alpha=1}^{n} c_\alpha G(\|\boldsymbol{x} - \boldsymbol{t}_\alpha\|_{\boldsymbol{W}}^2)$$

$$\|\boldsymbol{x} - \boldsymbol{x}_i\|_{\boldsymbol{W}}^2 = (\boldsymbol{x} - \boldsymbol{x}_i)^T \boldsymbol{W}^T \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{x}_i)$$

Extensions:

Then how they works?

$$f^*(\boldsymbol{x}) = \sum_{\alpha=1}^{n} c_\alpha G(\|\boldsymbol{x} - \boldsymbol{t}_\alpha\|_{\boldsymbol{W}}^2)$$

Try to find C, G, and W to find the optimal expansion.

A straight-forward approach: Gradient Descent.

$$\frac{\partial H[f^*]}{\partial c_\alpha} = 0, \qquad \frac{\partial H[f^*]}{\partial \boldsymbol{t}_\alpha} = 0, \qquad \frac{\partial H[f^*]}{\partial \boldsymbol{W}} = 0,$$

$$\alpha = 1, \cdots, n.$$

Solutions:

- for the c_α

$$\frac{\partial H[f^*]}{\partial c_\alpha} = -2 \sum_{i=1}^{N} \Delta_i G(\|\boldsymbol{x}_i - \boldsymbol{t}_\alpha\|_W^2);$$

- for the centers t_α

$$\frac{\partial H[f^*]}{\partial c_\alpha} = 4c_\alpha \sum_{i=1}^{N} \Delta_i G'(\|\boldsymbol{x}_i - \boldsymbol{t}_\alpha\|_W^2)\, \boldsymbol{W}^T\boldsymbol{W}(\boldsymbol{x}_i - \boldsymbol{t}_\alpha)$$

- and for $\boldsymbol{W}$

$$\frac{\partial H[f^*]}{\partial \boldsymbol{W}} = -4\boldsymbol{W} \sum_{\alpha=1}^{N} c_\alpha \sum_{i=1}^{N} \Delta_i G'(\|\boldsymbol{x}_i - \boldsymbol{t}_\alpha\|_W^2) Q_{i,\alpha}$$

Interpretation:

Getting closer to data samples:

The sum of the product of error and its activation value.

for the c_α

$$\frac{\partial H[f^*]}{\partial c_\alpha} = -2 \sum_{i=1}^{N} \Delta_i G(\|\boldsymbol{x}_i - \boldsymbol{t}_\alpha\|_W^2);$$

Interpretation:

Clustering:

Move the centers toward to the majority of the data.

- for the centers t_α

$$\frac{\partial H[f^*]}{\partial c_\alpha} = 4c_\alpha \sum_{i=1}^{N} \Delta_i G'(\|x_i - t_\alpha\|_W^2)\, W^T W(x_i - t_\alpha)$$

And when the W is identity matrix, t is just the weighted average of the data points.

Interpretation:

Again, PCA:

Converging the W with rows that are close to eigenvectors of Q(correlation matrix examples relative with t) with smallest eigenvalues.

- and for $\boldsymbol{W}$

$$\frac{\partial H[f^*]}{\partial \boldsymbol{W}} = -4\boldsymbol{W} \sum_{\alpha=1}^{N} c_\alpha \sum_{i=1}^{N} \Delta_i G'(\|\boldsymbol{x}_i - \boldsymbol{t}_\alpha\|_{\boldsymbol{W}}^2) Q_{i,\alpha}$$

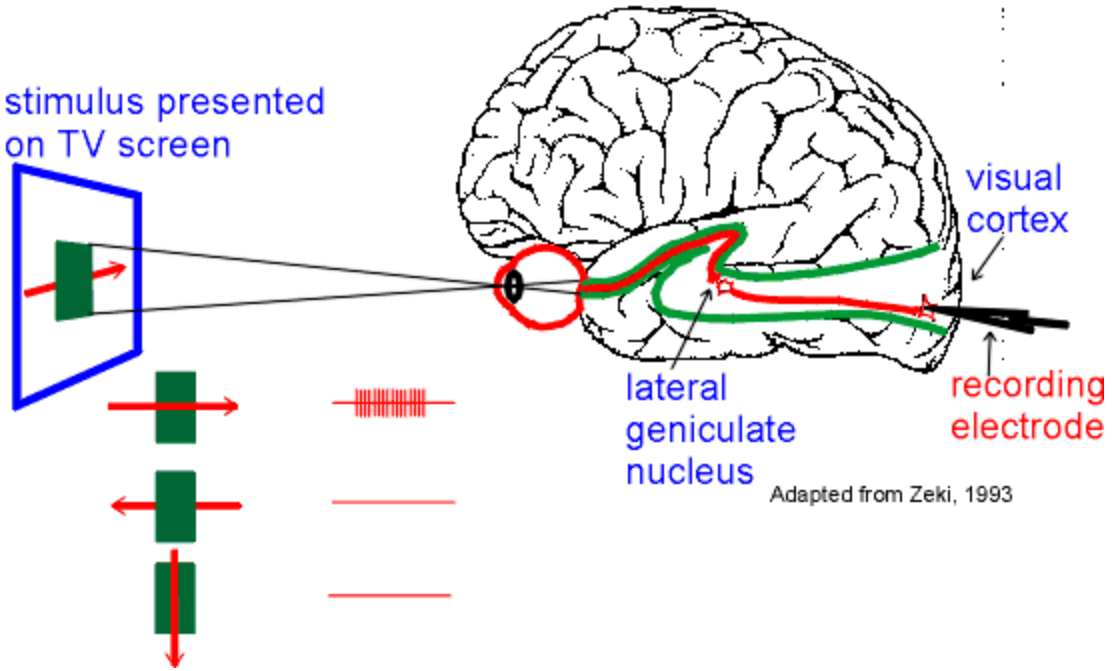
In other words, converge rows of W that span the space orthogonal to the space spanned by the principal components of the inputs.

More Improvement: Initial States(RBF)

1. Set the rows of W to be the vectors orthogonal to the eigenvectors with largest eigenvalues;
2. Set the centers to the center positions to a subset of the examples;
3. Directly find the c by using pseudo-inversion;

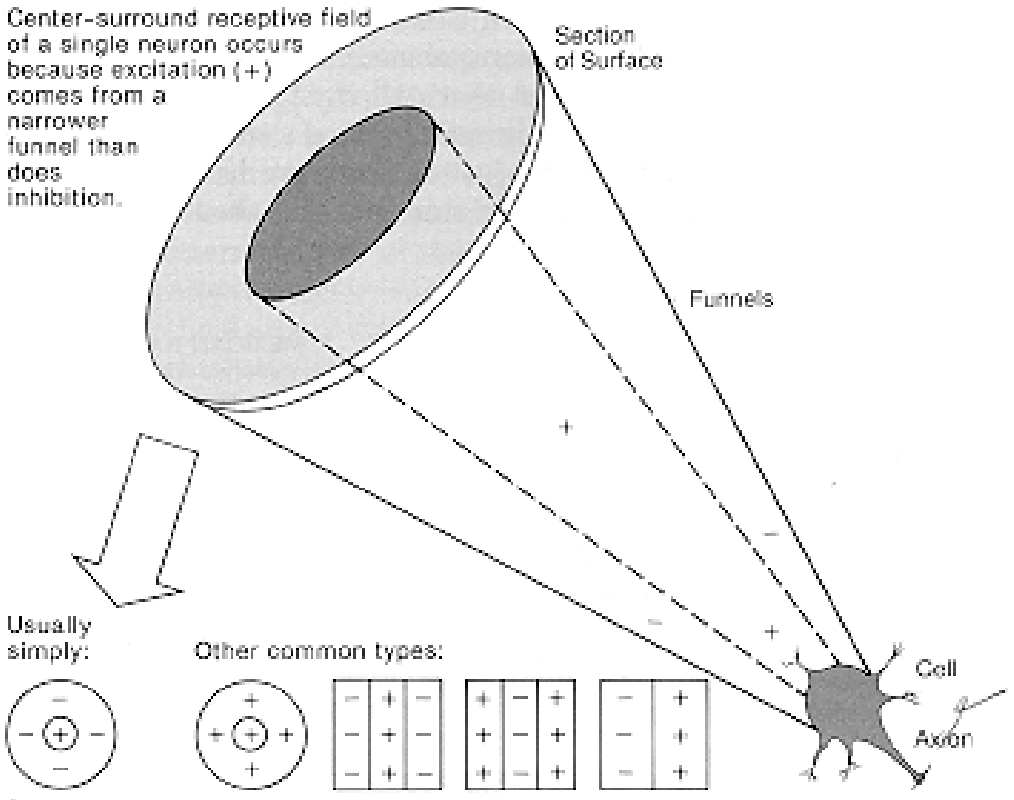
Interesting stuff
Neurobiology implications.

Are neurons calculating the same thing $G(||x-t||^2)$?



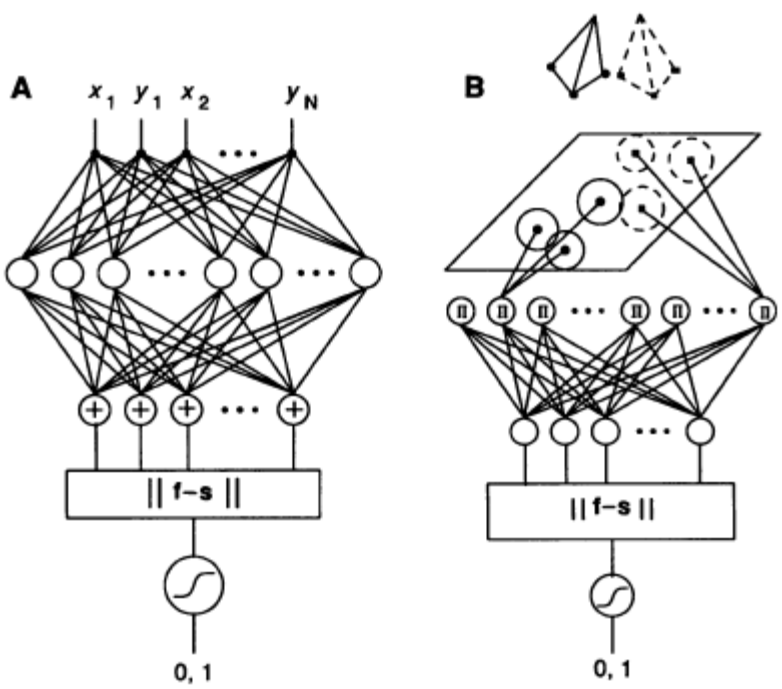
Interesting stuff
Neurobiology implications.

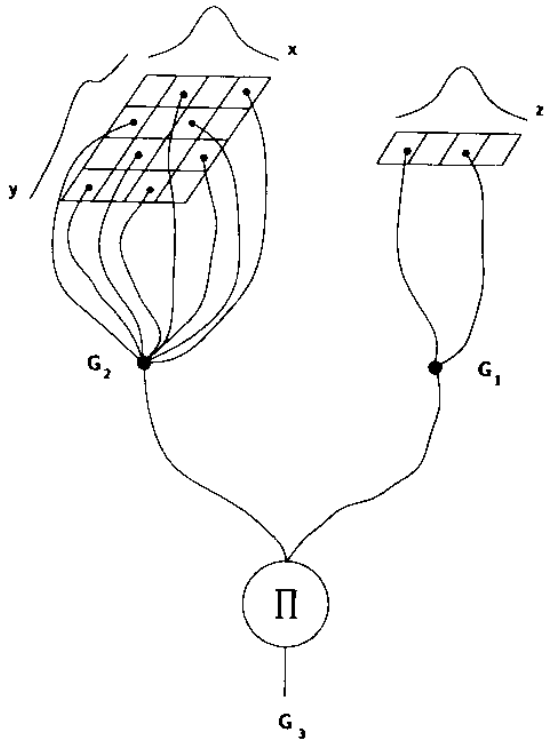
Receptive field:



Interesting stuff
Neurobiology implications.

Artificial Gaussian receptive fields on sensor array, transduce the implicit position to a number.



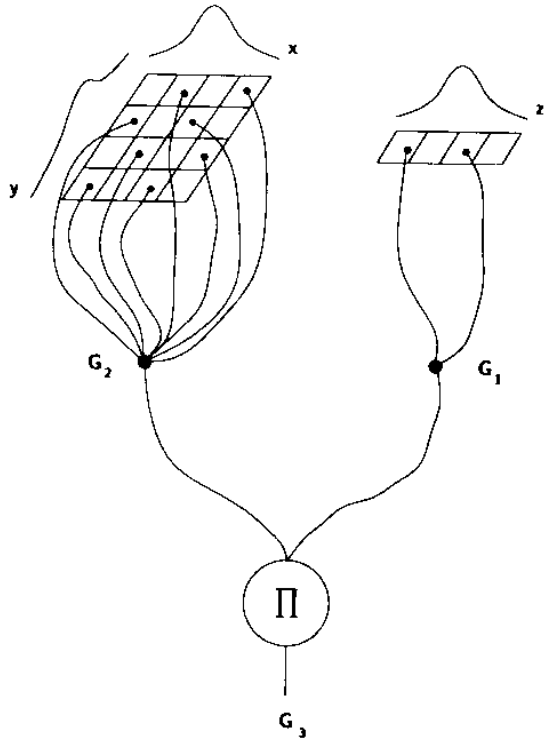


Interesting stuff

Neurobiology implications.

.

1. Each Gaussian function shows the “feature” in that patter.
2. And logically calculate all of the output “features” (And, Or, etc).



Biological plausibility.

Receptive Field, very similar structure: combinations of receptive field from 2D retinotopic arrays, somewhat similar to template-like cell.

However, actually, because of the lateral connections, the behaviors are different in lots of ways.

It is not too implausible for brain to adapt similar process. moving the centers, determining the weights, etc. which implies some mechanism for plasticity of brain?

Conclusion & Questions

1. Does the function approximation covers reinforcement learning, and even self-supervised learning scheme.
2. Does our brain do things beyond that?

3. Connection between regularization theory, Bayesian inference, MAP, complexity, and even the compression relations between them.
4. Relatively plausible computable neuroscience experiment based on receptive field evidence in neuroscience.