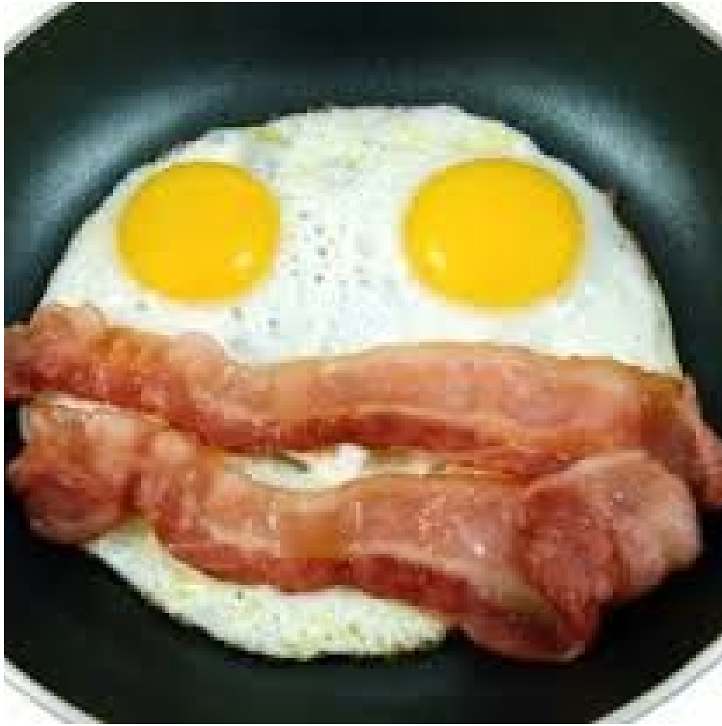
Heuristic Problem Solving

Suggestions and tools

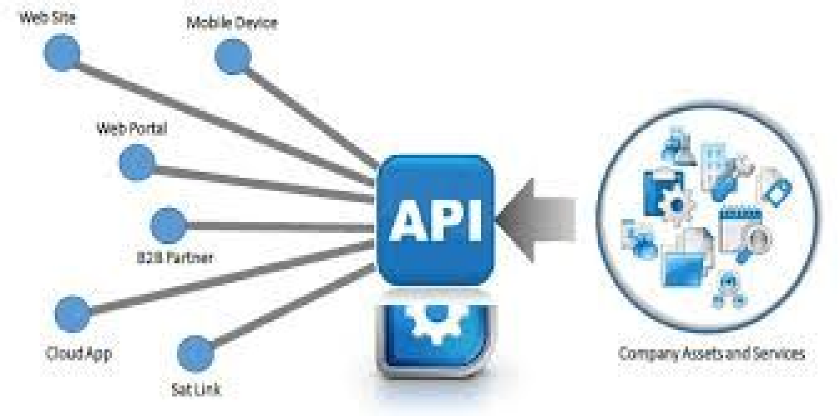
Are you Involved or Committed?

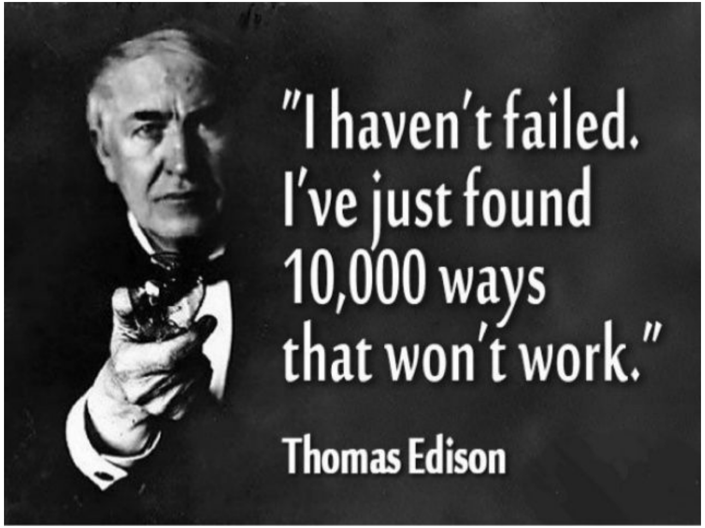


Skills to practice

- Form a team of 2 members
- Roles: interface, strategy and tactics
- Be coding all the time

Interface, Tactics and Strategy



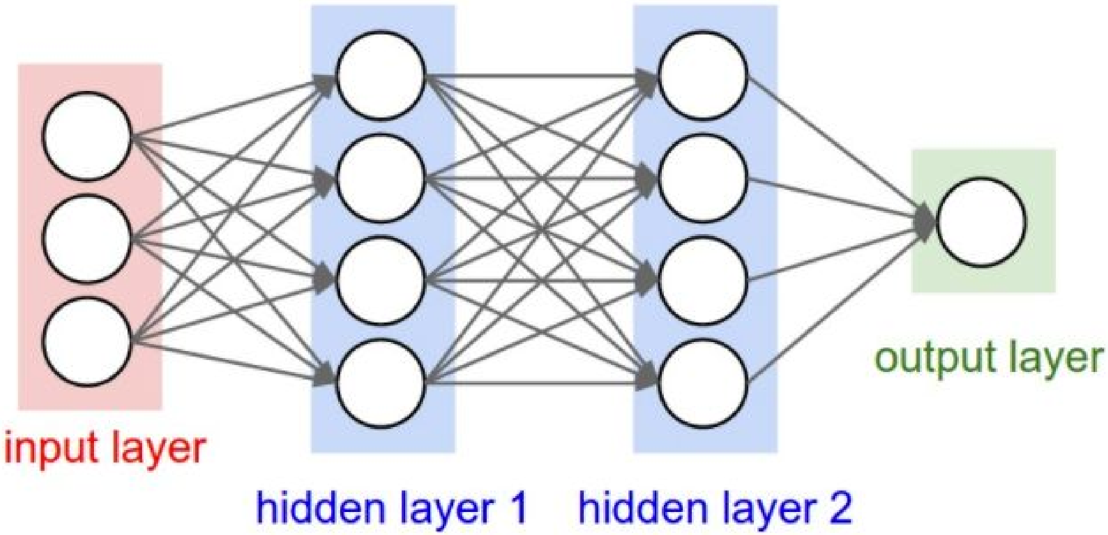


- **Rapid Prototype**
- **Fail fast, early and often**
- **Win, if you must**
- **Definitely, learn and have fun**

Good Luck

“Was mich nicht umbringt macht mich haerter” - Nietzsche

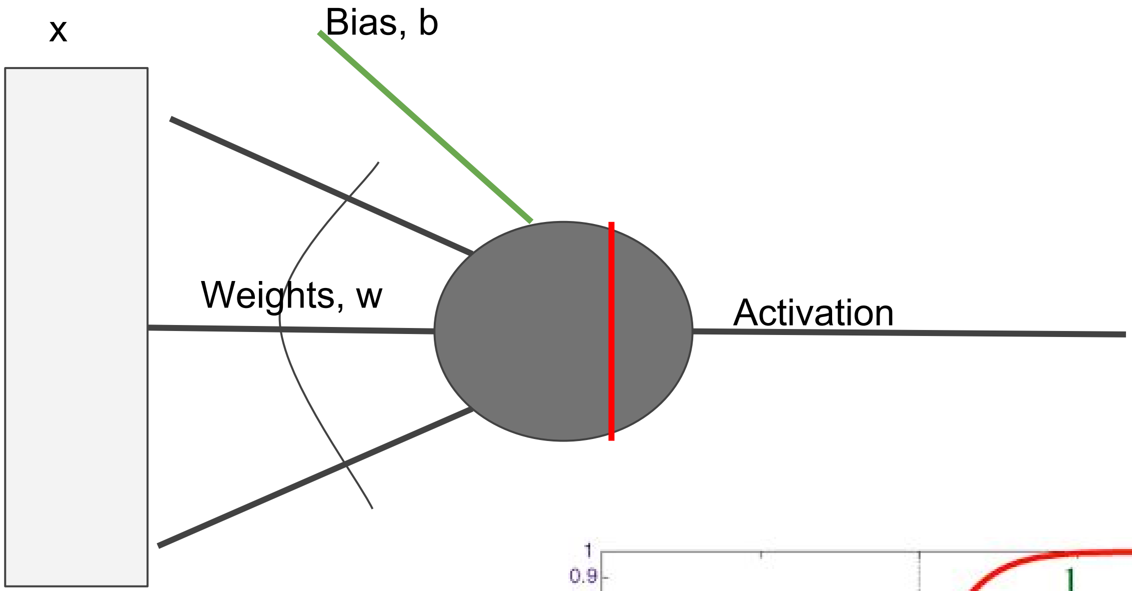
Neural Network at Light Speed (MLP)

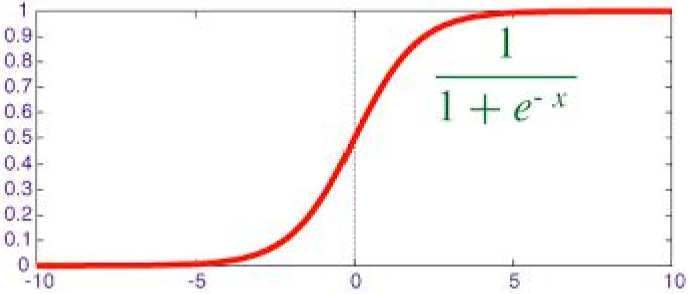


From CS231N Stanford

Neural Nets at Light Speed

Artificial Neuron

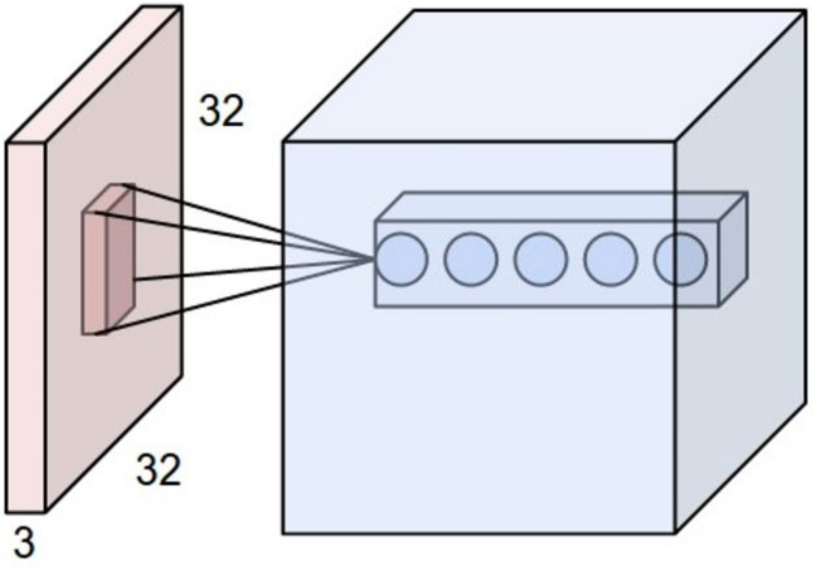


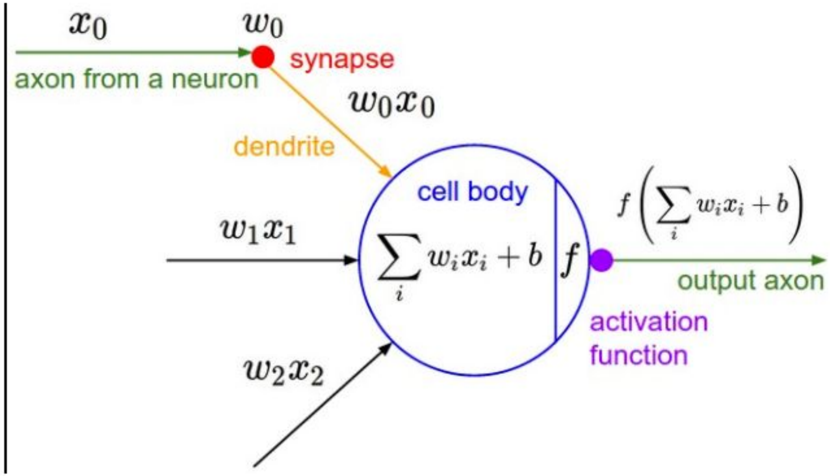


Linear, $z = wx + b$
Activation, $a = \sigma(z)$

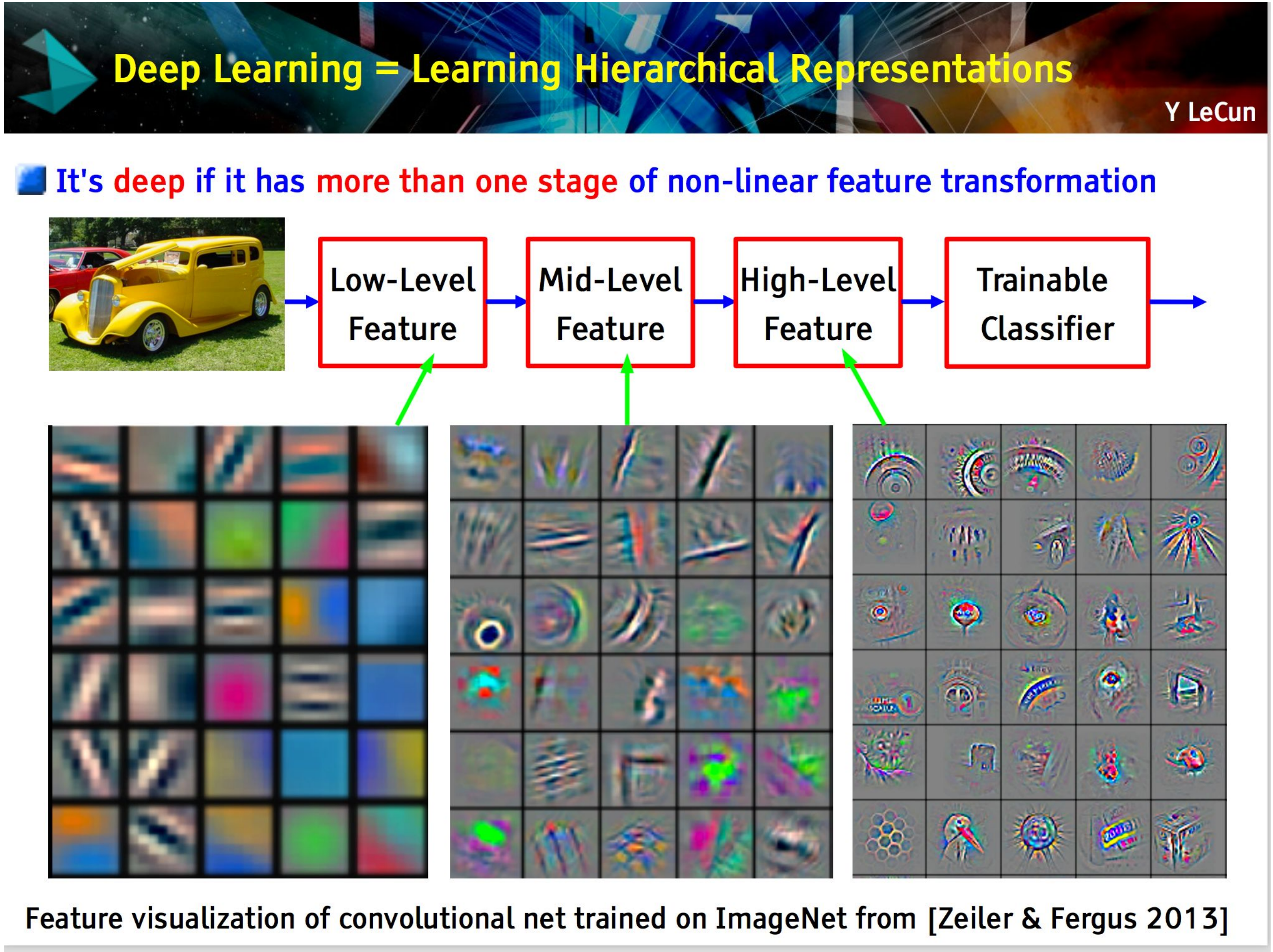
Backpropagation:
$\partial\mathcal{L}/\partial w = (\partial\mathcal{L}/\partial z)\,(\partial z/\partial w)$

Convolution



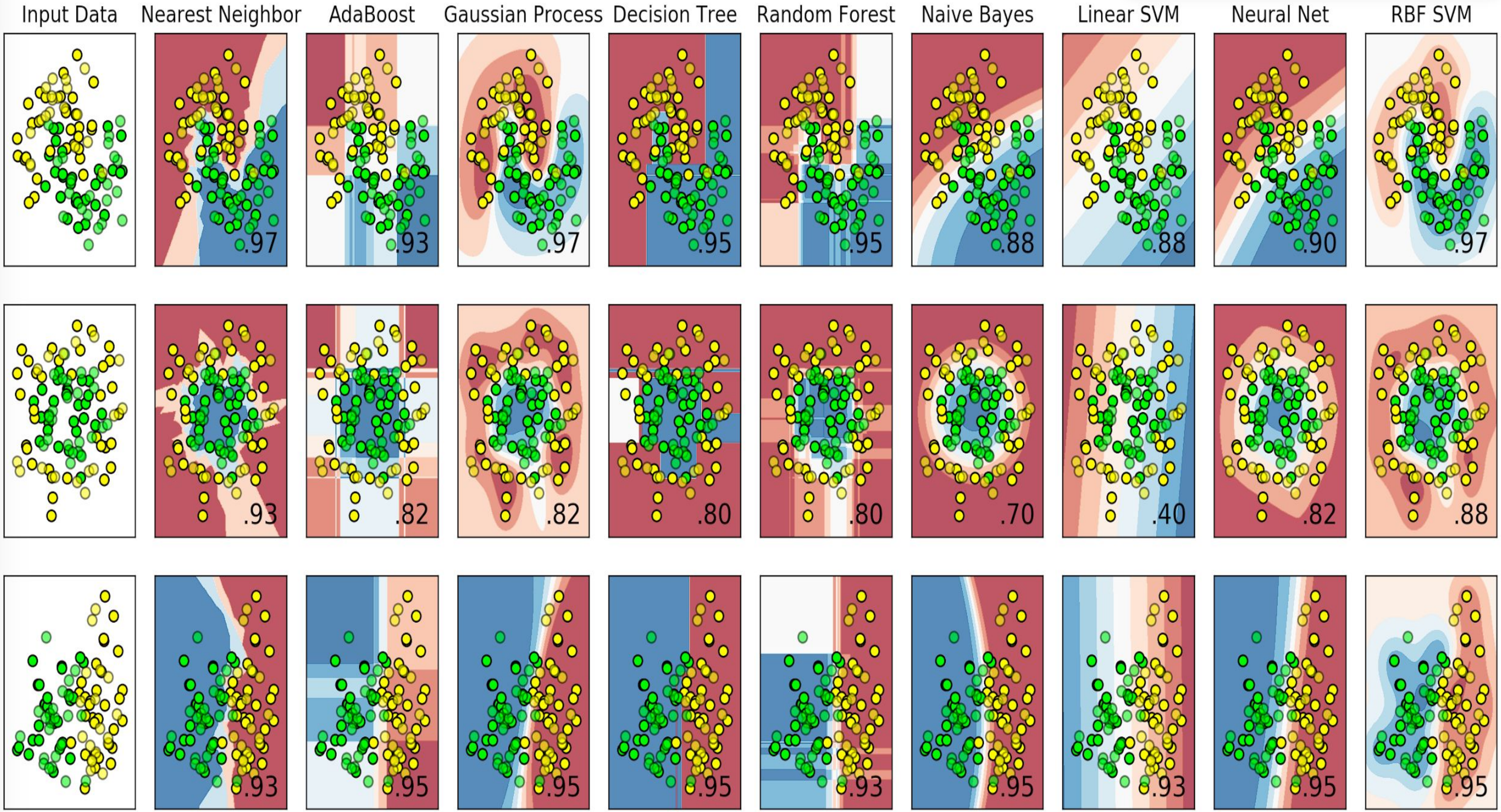


From CS231N Stanford



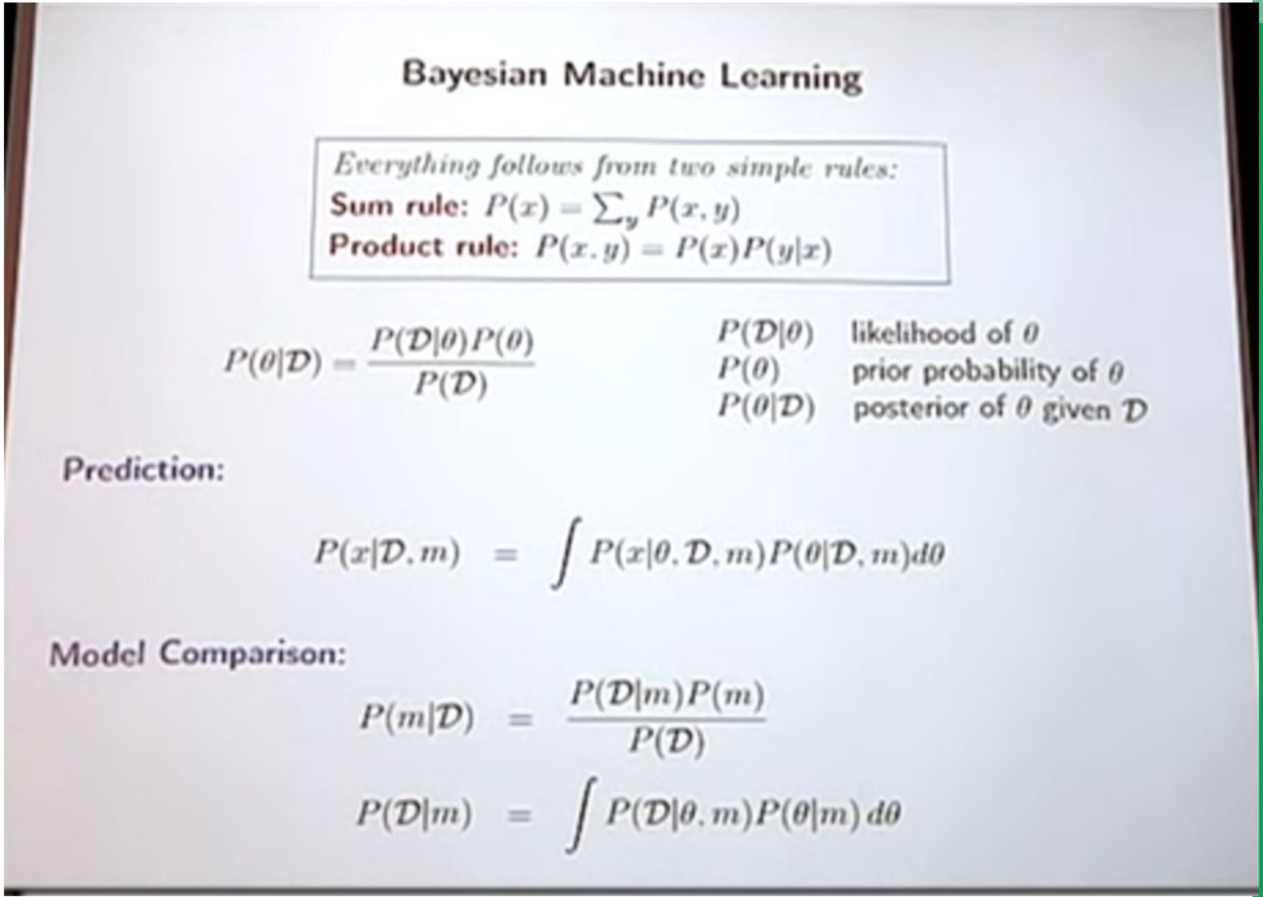
Features Learnt by Layers of a Neural Net

Quick Demo


Input Data
Nearest Neighbor
AdaBoost
Gaussian Process
Decision Tree
Random Forest
Naive Bayes
Linear SVM
Neural Net
RBF SVM
.97
.93
.97
.95
.95
.88
.88
.90
.97
.93
.82
.82
.80
.80
.70
.40
.82
.88
.93
.95
.95
.95
.93
.95
.93
.95
.95

Bayesian Learning

- Integration, not optimization
- Prediction is a convolution



By Z. Ghaharamani

Multivariate Gaussian Theorem (see KPM)

Theorem 4.2.1 (Marginals and conditionals of an MVN). *Suppose* $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$ *is jointly Gaussian with parameters*

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}, \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{pmatrix} \tag{4.12}$$

Then the marginals are given by

$$\begin{aligned} p(\mathbf{x}_1) &= \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}) \\ p(\mathbf{x}_2) &= \mathcal{N}(\mathbf{x}_2 | \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22}) \end{aligned}$$

and the posterior conditional is given by

$$\begin{aligned} p(\mathbf{x}_1 | \mathbf{x}_2) &= \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2}) \\ \boldsymbol{\mu}_{1|2} &= \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{11}^{-1}\boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\Sigma}_{1|2}\left(\boldsymbol{\Lambda}_{11}\boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2)\right) \\ \boldsymbol{\Sigma}_{1|2} &= \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \boldsymbol{\Lambda}_{11}^{-1} \end{aligned}$$

By N. Freitas

Multivariate Gaussian

- Useful Properties
- Joint leads to Marginal and Conditional

Gaussian process covariance functions (kernels)

$p(f)$ is a Gaussian process if for *any* finite subset $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the marginal distribution over that finite subset $p(\mathbf{f})$ has a multivariate Gaussian distribution.

Gaussian processes (GPs) are parameterized by a mean function, $\mu(x)$, and a covariance function, or kernel, $K(x, x')$.

$$p(f(x), f(x')) = \mathrm{N}(\mu, \Sigma)$$

where

$$\mu = \begin{bmatrix} \mu(x) \\ \mu(x') \end{bmatrix} \quad \Sigma = \begin{bmatrix} K(x,x) & K(x,x') \\ K(x',x) & K(x',x') \end{bmatrix}$$

and similarly for $p(f(x_1), \ldots, f(x_n))$ where now μ is an $n \times 1$ vector and Σ is an $n \times n$ matrix.

By Z. Ghaharamani

Gaussian Process

- Mean function
- Kernel/covariance function

Gaussian process covariance functions

Gaussian processes (GPs) are parameterized by a mean function, $\mu(x)$, and a covariance function, $K(x, x')$.

An example covariance function:

$$K(x_i, x_j) = v_0 \exp\left\{-\left(\frac{|x_i - x_j|}{r}\right)^\alpha\right\} + v_1 + v_2 \delta_{ij}$$

with parameters $(v_0, v_1, v_2, r, \alpha)$

These kernel parameters are interpretable and can be learned from data:

v_0	signal variance
v_1	variance of bias
v_2	noise variance
r	lengthscale
α	roughness

Once the mean and covariance functions are defined, everything else about GPs follows from the basic rules of probability applied to mutivariate Gaussians.

By Z. Ghaharamani

GP learning the kernel

Consider the covariance function K with hyperparameters $\theta = (v_0, v_1, r_1, \ldots, r_d, \alpha)$:

$$K_\theta(\mathbf{x}_i, \mathbf{x}_j) = v_0 \exp\left\{ -\sum_{d=1}^{D} \left(\frac{|x_i^{(d)} - x_j^{(d)}|}{r_d} \right)^{\alpha} \right\} + v_1$$

Given a data set $\mathcal{D} = (\mathbf{X}, \mathbf{y})$, how do we learn θ?

The marginal likelihood is a function of θ

$$p(\mathbf{y}|\mathbf{X}, \theta) = \mathcal{N}(\mathbf{0}, \mathbf{K}_\theta + \sigma^2 \mathbf{I})$$

where its log is:

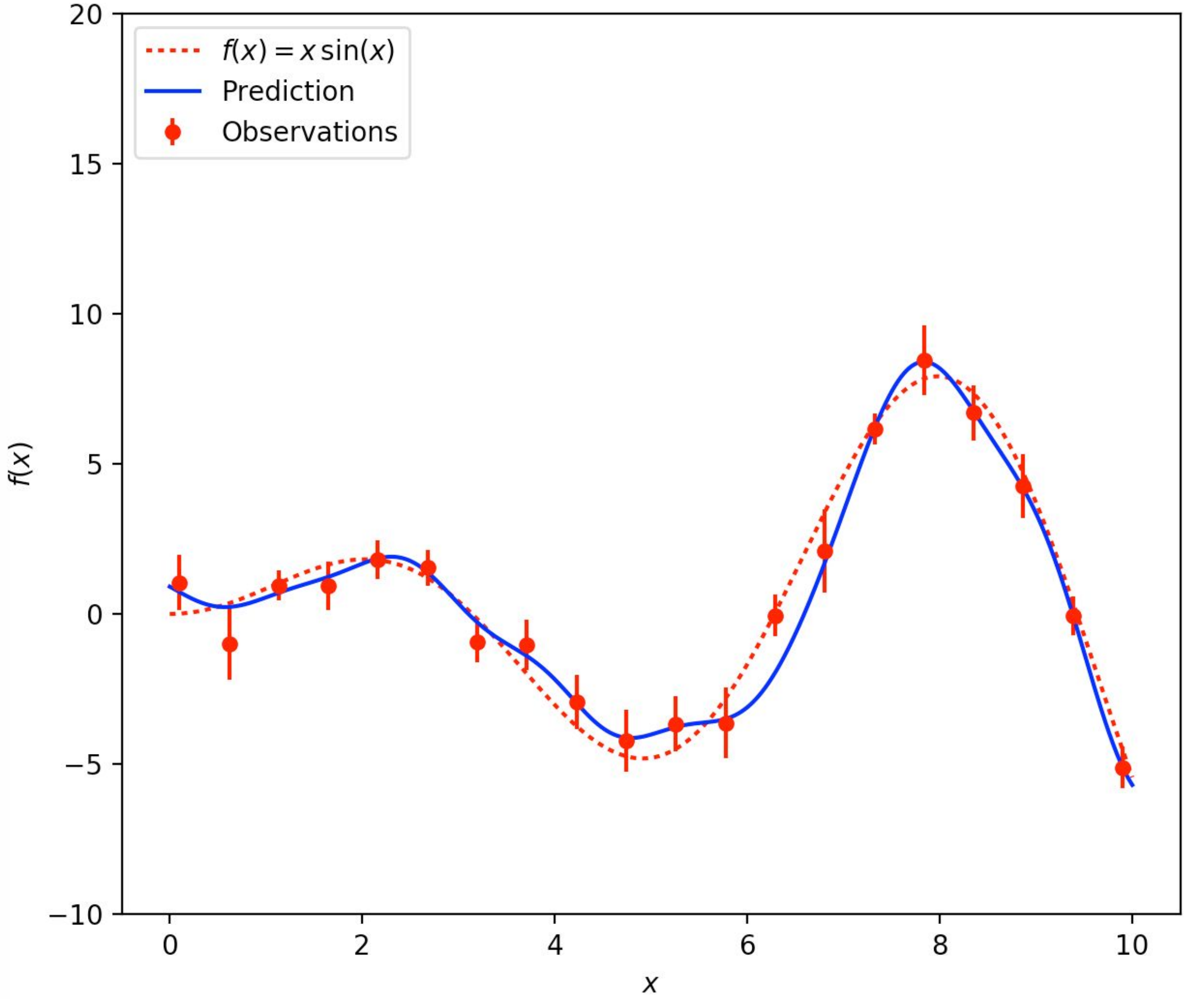
$$\ln p(\mathbf{y}|\mathbf{X}, \theta) = -\frac{1}{2} \ln \det(\mathbf{K}_\theta + \sigma^2 \mathbf{I}) - \frac{1}{2} \mathbf{y}^\top (\mathbf{K}_\theta + \sigma^2 \mathbf{I})^{-1} \mathbf{y} + \text{const}$$

which can be optimized as a function of θ and σ.

Alternatively, one can infer θ using Bayesian methods, which is more costly but immune to overfitting.

Learning the Kernel

By Z. Ghaharamani



Gaussian Regressor

Highly effective

Simple and easy

Parametric to Non-parametric

Examples of non-parametric models

Bayesian nonparametrics has many uses.

Parametric	Non-parametric	Process	Application
polynomial regression	Gaussian processes	GP	function approx.
logistic regression	Gaussian process classifiers	GP	classification
mixture models, k-means	Dirichlet process mixtures	DP / CRP	clustering
hidden Markov models	infinite HMMs	HDP	time series
factor analysis/pPCA/PMF	infinite latent factor models	BP / IBP	feature discovery

By Z. Ghaharamani