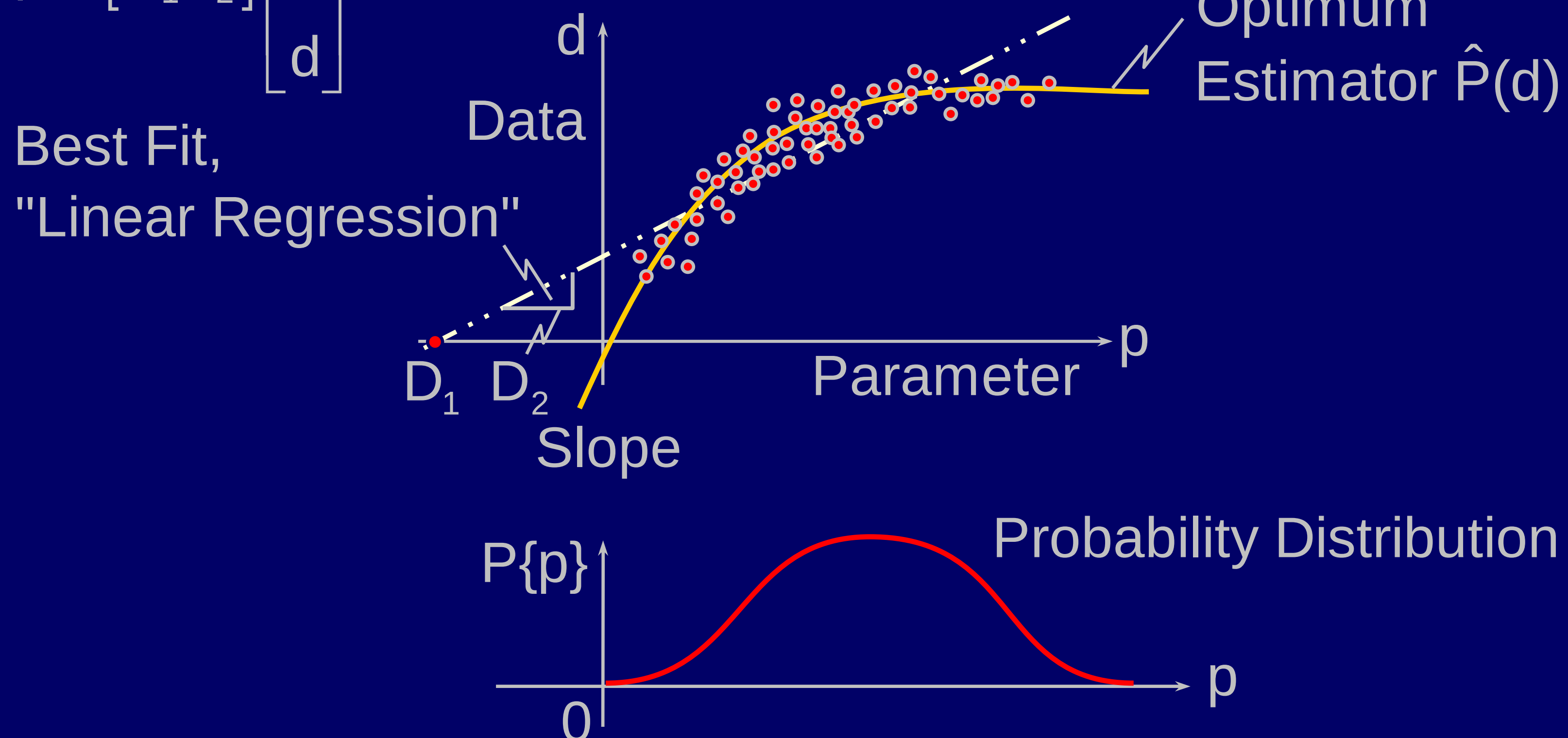
Nonlinear Estimation

Professor David H. Staelin
Massachusetts Institute of Technology

Case I: Nonlinear Physics

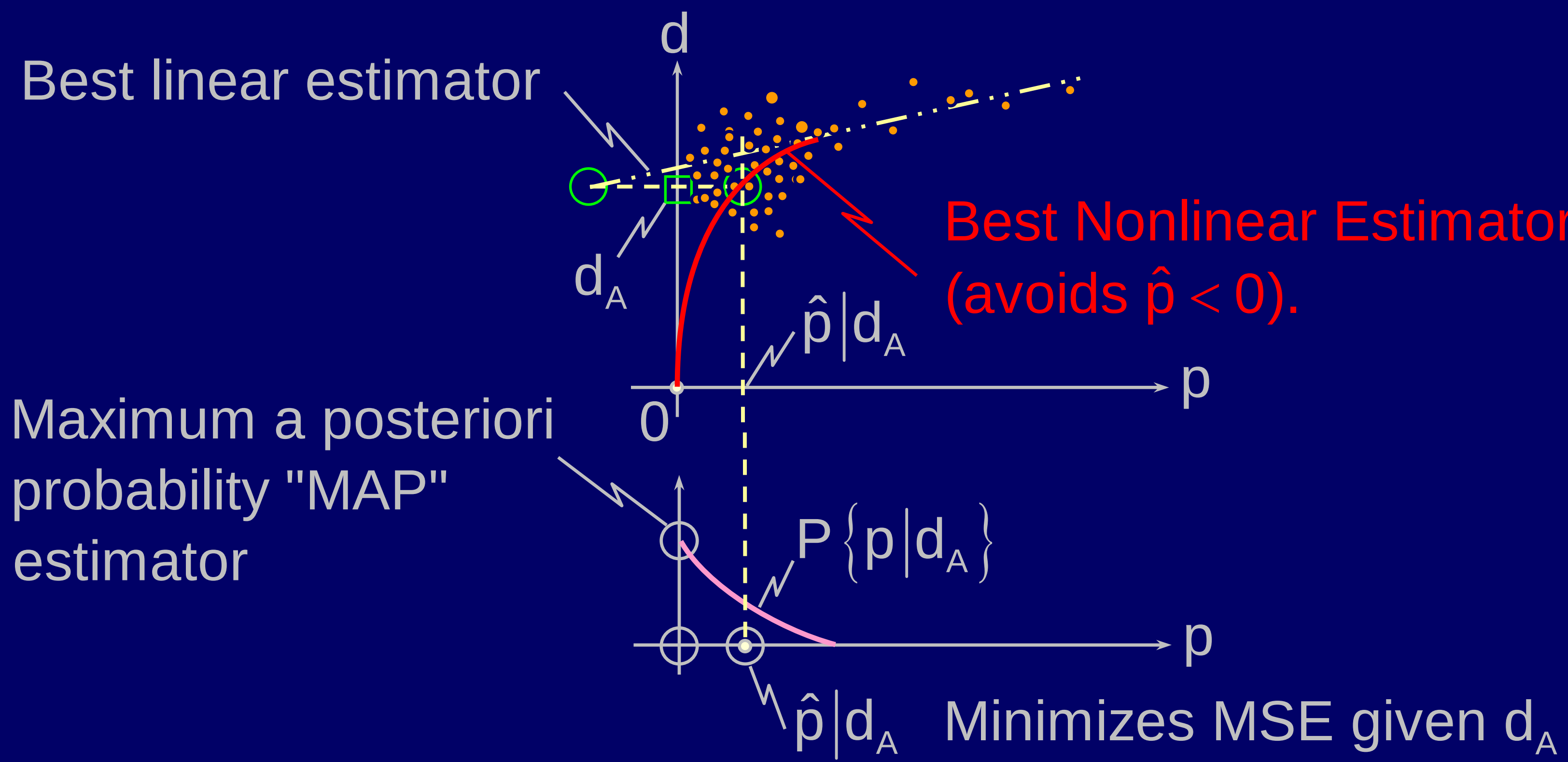
$$\hat{p} = [D_1 D_2]\begin{bmatrix} 1 \\ d \end{bmatrix}$$



Minimum Square Error (MSE) is usually the "optimum" sought.

It is important that the "training" data used for regression is similar to, but independent of, the "test" data used for performance evaluation.

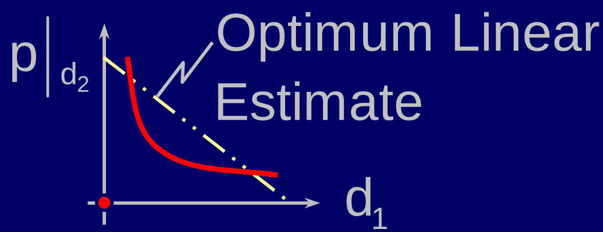
Case II: Non-Gaussian Statistics

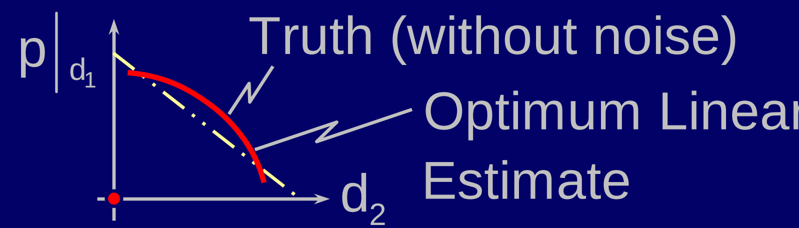


Note: Linear estimator can set $\hat{p}<0$ (which is non-physical):
nonlinear estimator approaches correct asymptotes as $d \to \pm\infty$

Note: Physics here is linear. Can use polynomials etc. for
$\hat{p}(d)$, or recursive linear estimators updating $\underline{\underline{D}}$.

Linear Estimates for Nonlinear Problems





Linear estimates are suboptimum for d_1 or d_2 alone.

Say $\quad d_1 = a_0 + a_1 p + a_2 p^2$ and $d_2 = b_0 + b_1 p + b_2 p^2$

Then $\quad p^2 = \left(d_2 - b_0 - b_1 p\right)/b_2$

$$d_1 = a_0 + a_1 p + a_2\left(d_2 - b_0 - b_1 p\right)/b_2 = c_0 + c_1 p + c_2 d_2$$

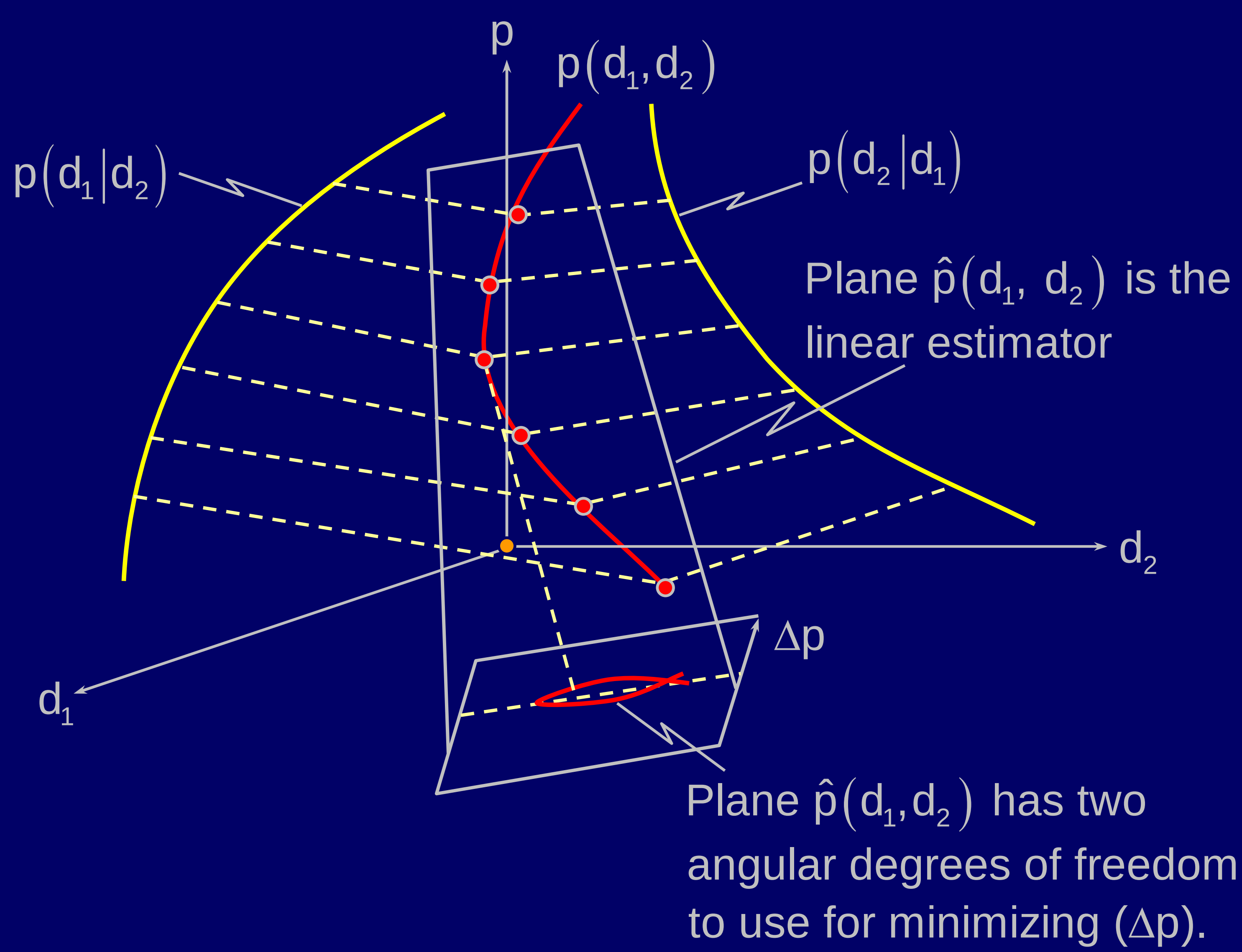
(defines plane in $\left(p,\ d_1,\ d_2\right)$- space

Therefore $\hat{p} = p = \left(-c_0 + d_1 - c_2 d_2\right)/c_1$ where $c_1 = a_1 - \dfrac{a_2 b_1}{b_2} \neq 0$

$\hat{p}\left(d_1,\ d_2\right)$ can be linear in d_1, d_2 and (noiseless case) perfect, even though $p\left(d_1,\ d_2\right)$ is nonlinear

In general, the more varied the data $\bar{d}$, the smaller the performance gap between linear estimators and nonlinear ones, which are a superset.

Linear Estimates for Nonlinear Problems



Generalization to Nth-Order Nonlinearities

Let

$$d_1 = c_1 + a_{11}p + a_{12}p^2 + \ldots + a_{1n}p^n$$

$$d_2 = c_2 + a_{21}p + a_{22}p^2 + \ldots + a_{2n}p^n$$

$$\vdots$$

$$d_n = c_n + a_{n1}p + a_{n2}p^2 + \ldots + a_{nn}p^n$$

where $d_1, d_2, \ldots, d_n$ are observed noise-free data related to p by an nth-order polynomial.

Claim: In non-singular cases there exists an exact linear estimator $\hat{p} = \bar{\bar{D}}\,\bar{d} + \text{constant}$.

Note: For perfect linear estimation the number of different observations must equal or exceed n, the maximum order of the polynomials characterizing the physics.

Nonlinear Estimators for Nonlinear Problems

Approaches: 1) $\bar{\hat{p}} = \bar{\bar{D}}\bar{d}_{aug}$ where $\bar{d}$ is augmented via polynomials

2) Rank reduction first via KLT

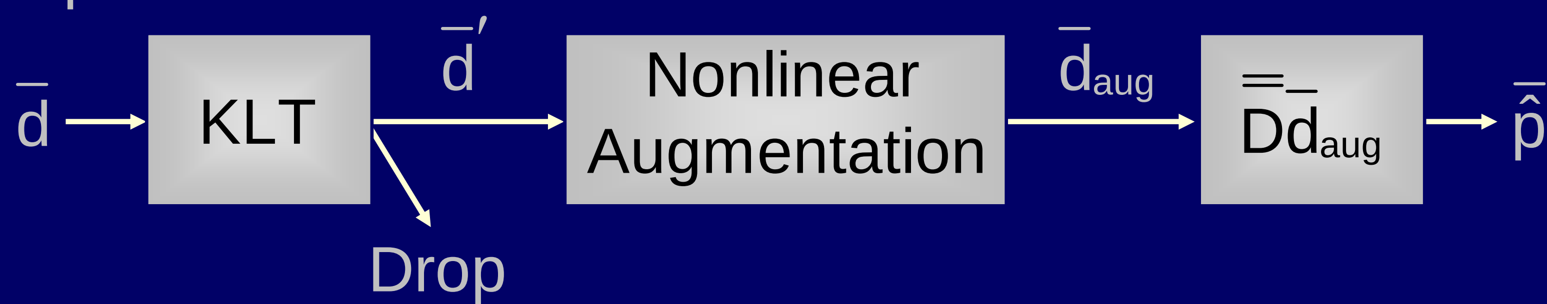
3) Neural nets

4) Iterations with $\bar{\bar{D}}(\hat{p})$

Case (1): $\bar{d}_{aug} = 1,\ d_1,\ d_2,\ d_1^2,\ d_2^2,\ d_1 d_2,\ d_1^3,$ for example

Case (2): Use of KLT to Reduce Rank

Option A:



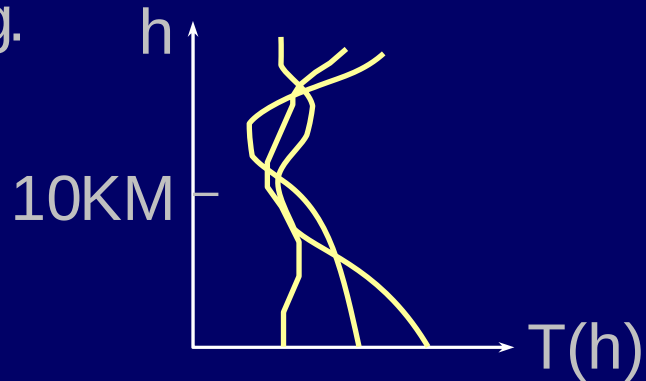
Useful if $\bar{d}$ is of High Order

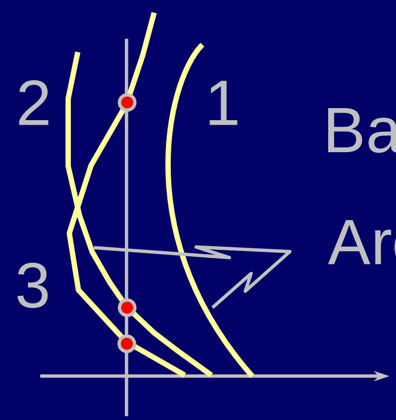
"Karhunen-Loeve Transform" (KLT)

$\bar{d}' \triangleq \bar{\bar{K}}\bar{d}$ transforms a noisy redundant data vector $\bar{d}$ into a [reduced-rank] vector $\bar{d}'$ with most variance in d'_1, least in d'_n, where $E\left[d'_i d'_j\right] = \delta_{ij}\lambda_i$, and $\lambda_i \geq \lambda_{j>i}$

$$\bar{\bar{C}}_{dd} \triangleq E\left[\bar{d}\bar{d}^t\right] = \bar{\bar{K}}^t \begin{bmatrix} \lambda_1 & \dots & 0 \\ \vdots & \lambda_2 & \vdots \\ 0 & \dots & \lambda_n \end{bmatrix} \bar{\bar{K}}$$

e.g.



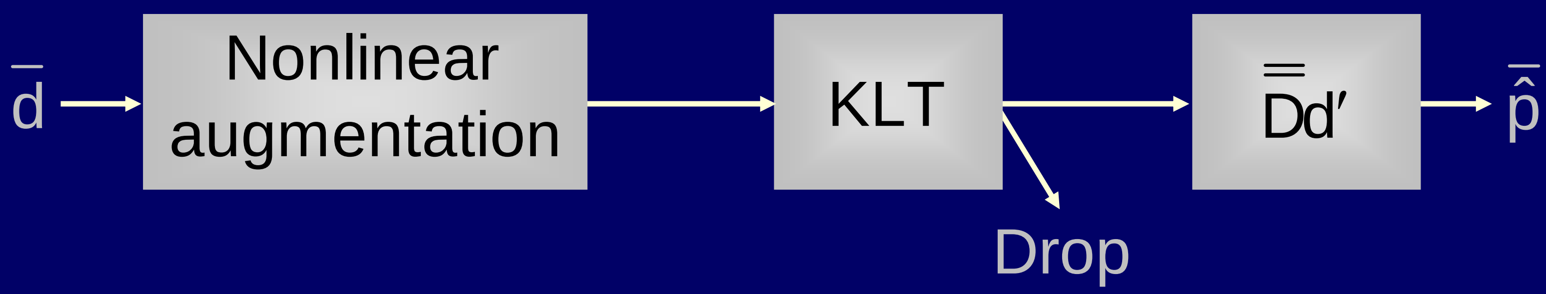


Basis Functions Are Rows of $\bar{\bar{K}}$

The KLT is essentially the same as Principal Component Analysis (PCA) and Empirical Orthogonal Functions (EOF).

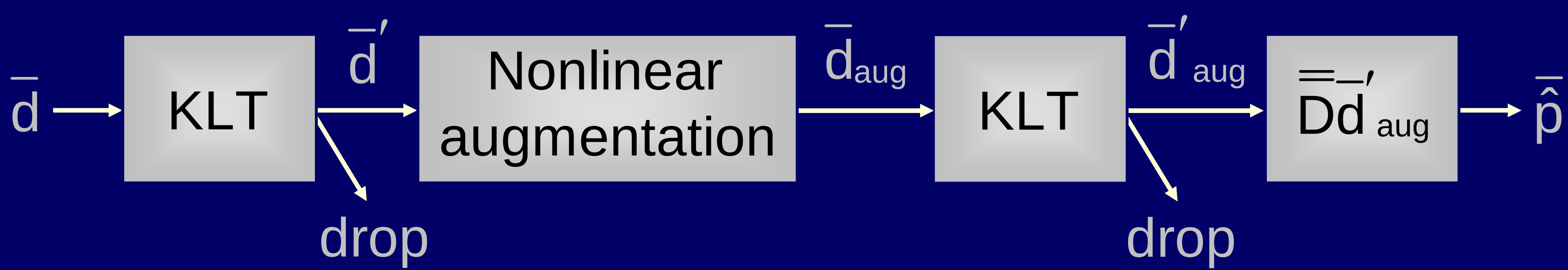
Nonlinear Estimators Using Rank Reduction

Option B:



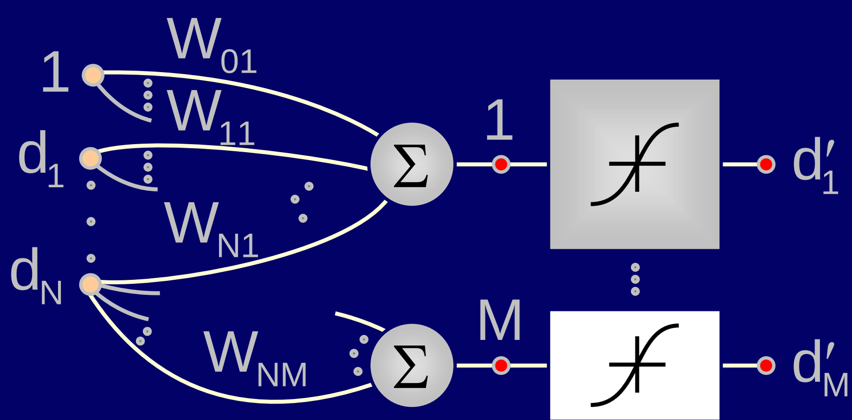
Useful if $\bar{d}$ is very nonlinear

Option C:

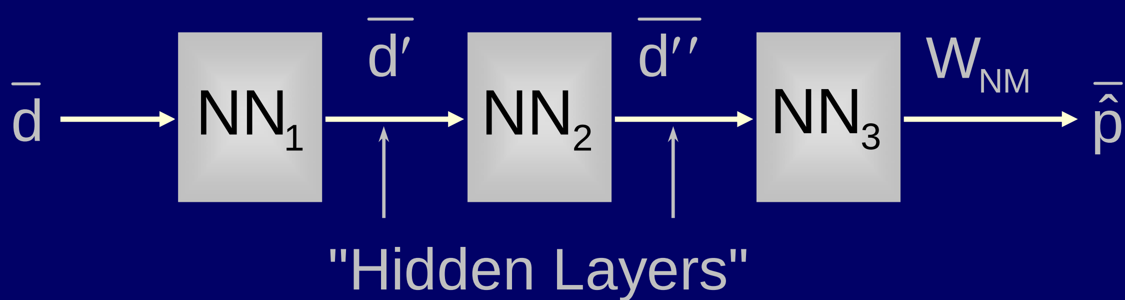


Neural Networks (NN)

Neural nets compute complex polynomials with great efficiency, simplicity.



NN coefficients computed via relaxation optimization back propagation.



NN may be recursive or purely Feed-Forward (FFNN)

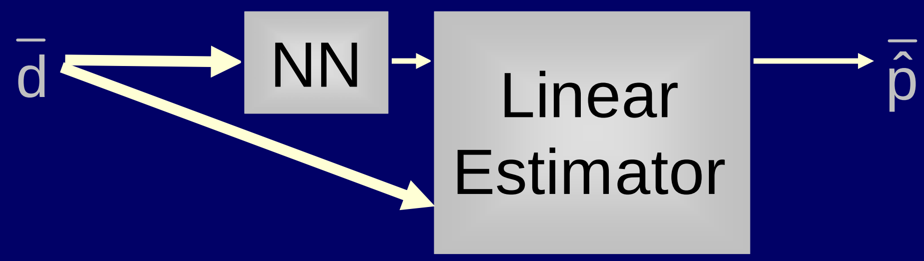
Neural Networks (NN)

Weights W_{ij} are found by guessing, then perturbing to reduce cost function on $\left[\hat{\bar{p}} - \bar{p}\right]$ over "training set" of known $\left[\bar{d}, \bar{p}\right]$. Training can be tedious. Commercial software tools are commonly used.

Degrees of freedom in training set should exceed the number of weights by a modest factor of $\sim$ 3-10. Risks of "overtraining" are reduced by monitoring NN performance on independent test data.

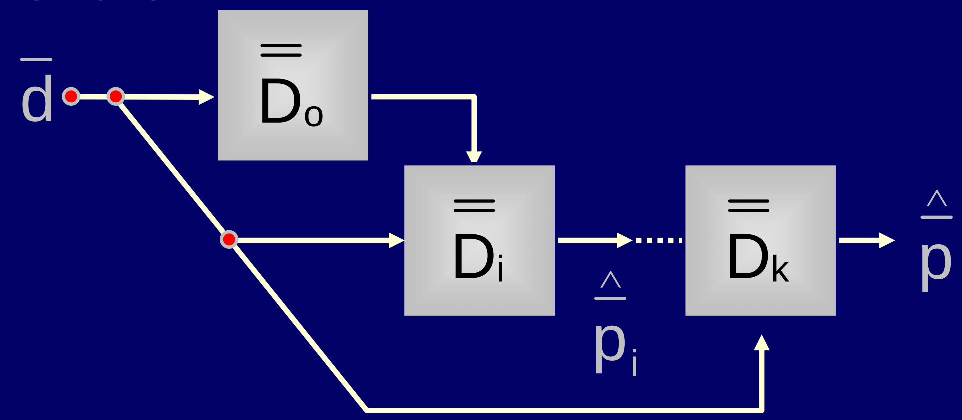
The more nonlinear problems need more layers and more internal nodes, as determined empirically. Neural nets can also perform recognition tasks.

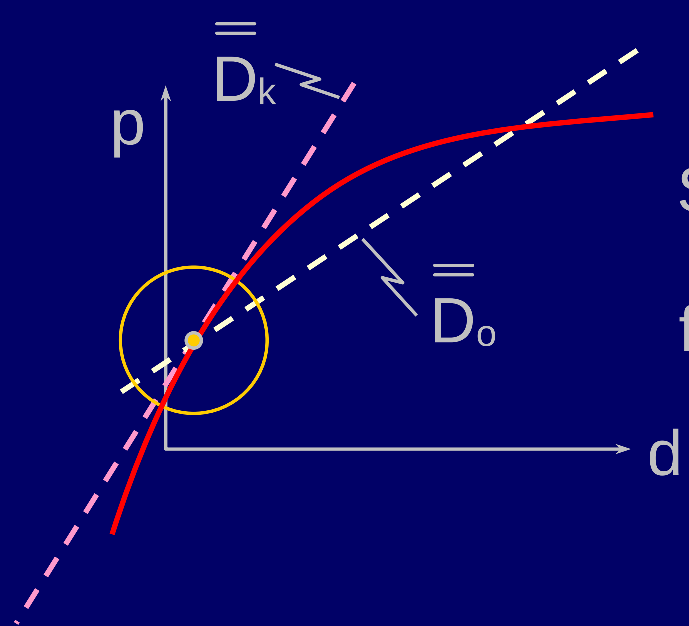
Simple neural nets are easier to train correctly, so merge with linear techniques, e.g.



Accommodating Nonlinearity via Iteration

A. Library Retrievals

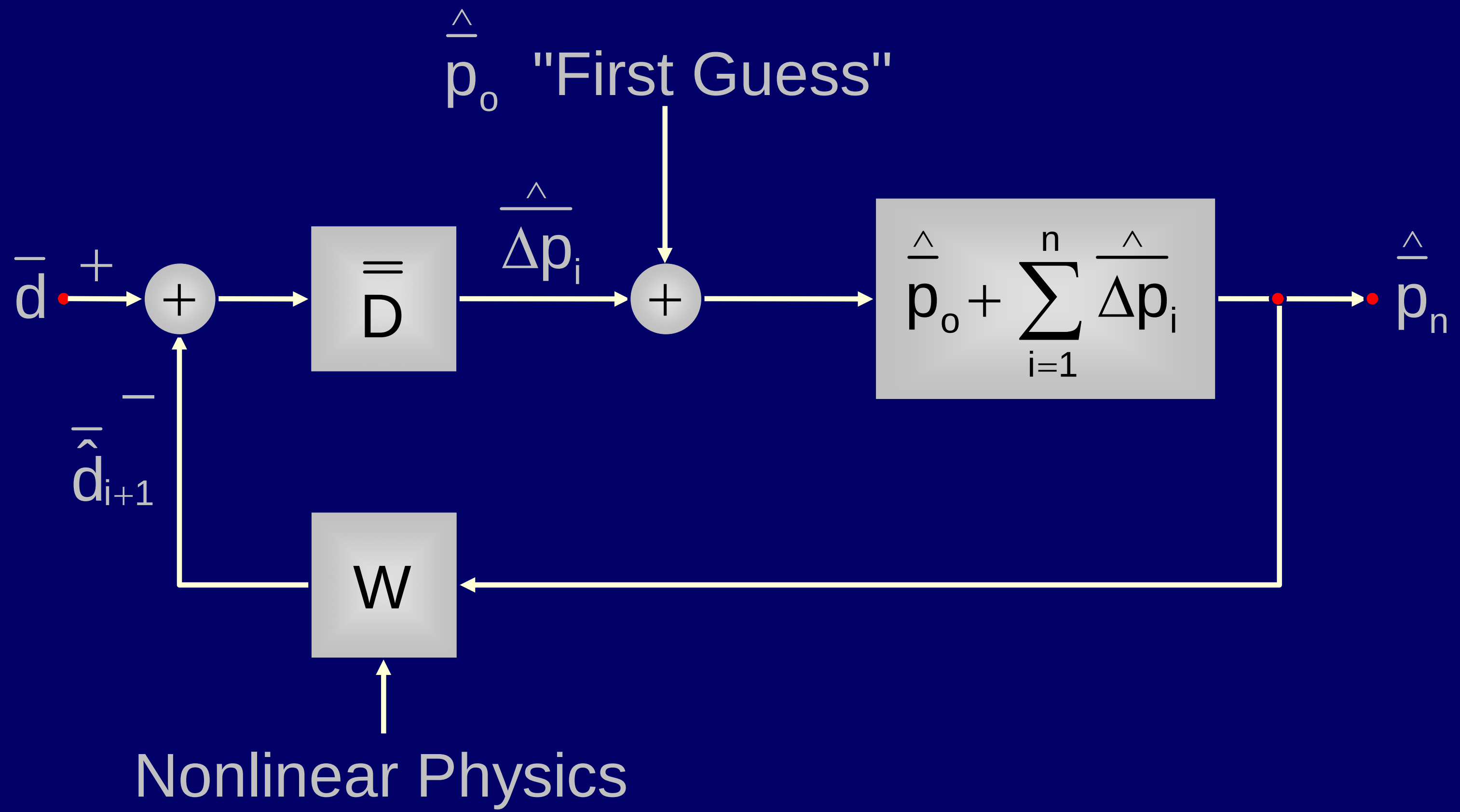




Successive $\bar{\bar{D}}_i$ Chosen from Library as $f\left(\hat{\underline{p}}_i\right)$

Accommodating Nonlinearity via Iteration

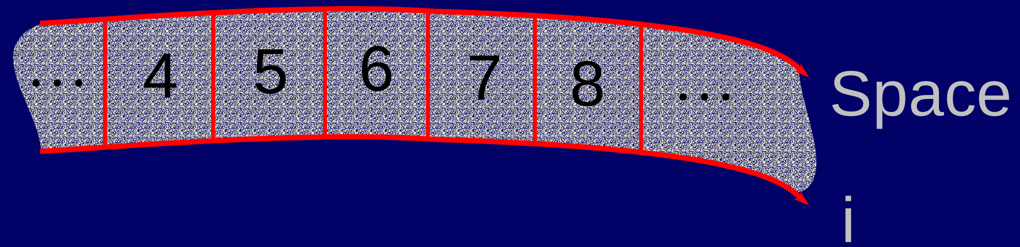
B. Physics Recomputation



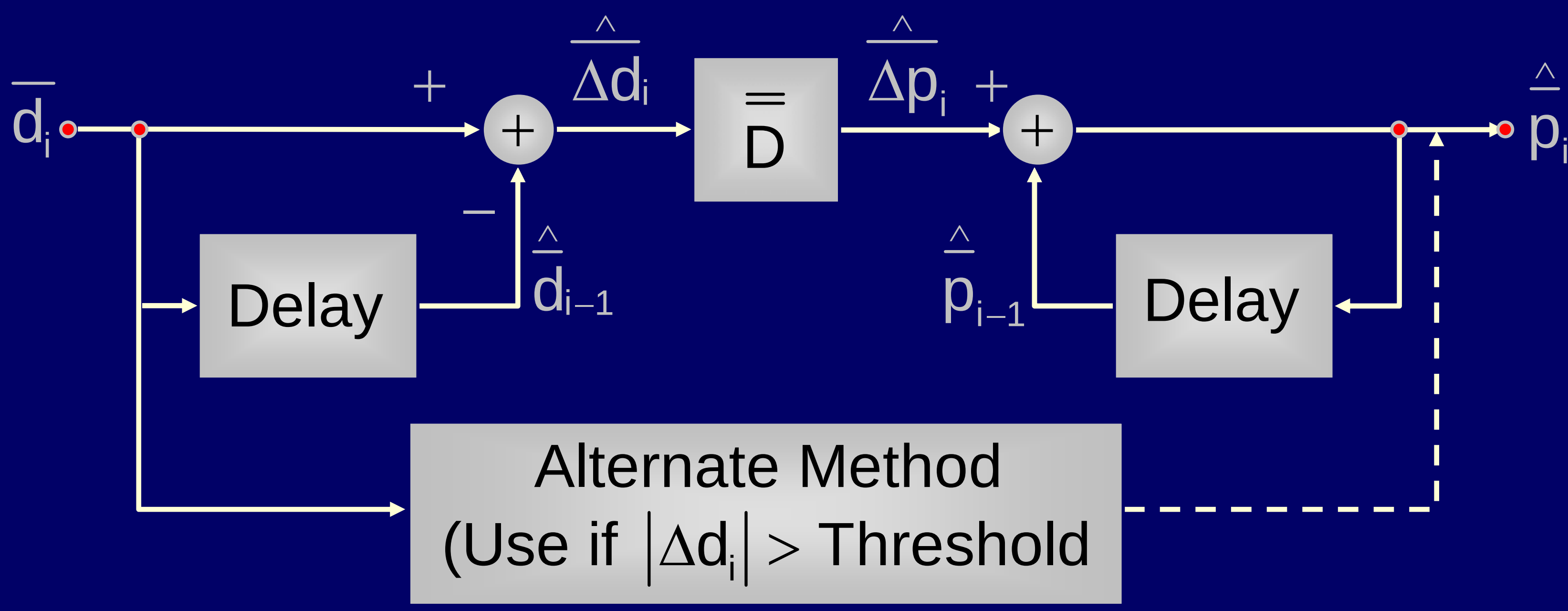
Essential only that $\bar{\bar{D}}$ always yields $\Delta\hat{\bar{p}}_i$ in right direction (so errors shrink).

Accommodating Nonlinearity via Iteration

C. Spatial Iteration



Works well if data is smooth.



Multi-Dimensional Estimation

e.g., Atmospheric Temperature Profiles

1) $\overline{\hat{T}}(h, x, y) = f\left(\overline{d}(x, y)\right)$

2) $\overline{\hat{T}}(h, x, y) = f\left(\overline{d}(x, y), \overline{d}(x \pm \Delta, y \pm \Delta), \ldots\right)$

Estimate improved by additional correlations and degrees of freedom.

Genetic Algorithms

1) Characterize algorithm by segmented character string, e.g., a binary number.
2) Significance of string is defined, e.g.,
   A. As weights in a linear estimator or neural net.
   B. It characterizes the architecture of a neural net.
3) Many algorithms (strings) are tested, and the metrics for each are evaluated for some ensemble of test cases.
4) Randomly combine elements of best algorithms to form new ones; some random changes may be made too.
5) Iterate steps (3) and(4) until asymmtotic optimum is reached. Can be used for pattern recognition, signal detection, parameter estimation, and other purposes.
6) Proper choice of "gene" size accelerates convergence (genes swapped as units).