

# P4-36

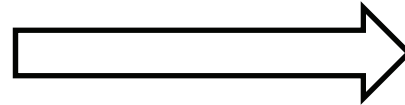
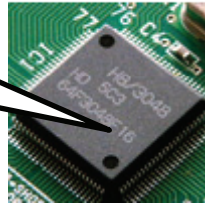
## Limited General Regression Neural Network

### による追記学習

山内康一郎(中部大学)

目的:リソースに制限のある組み込み機器向けの追記学習アルゴリズム構築を目指す。

追記学習  
アルゴリズム  
(one-pass  
Learning 可能)

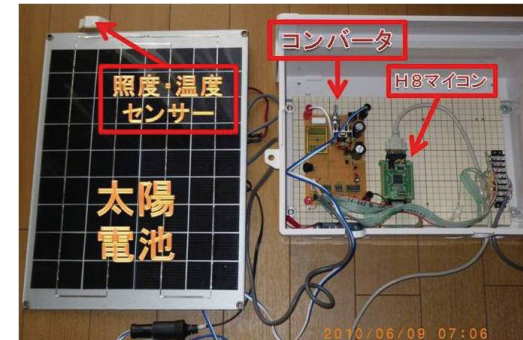


マイクロコンピュータ  
(※デスクトップコンピュータではない)

以下の条件が必要:

- 厳しいリソース制限があっても良い。
- 少ない計算量で実用的速度で計算できる。
- 非i.i.d.サンプル(非独立、経年変化するデータ)でも安定した学習結果が得られる。

想定される適用例:



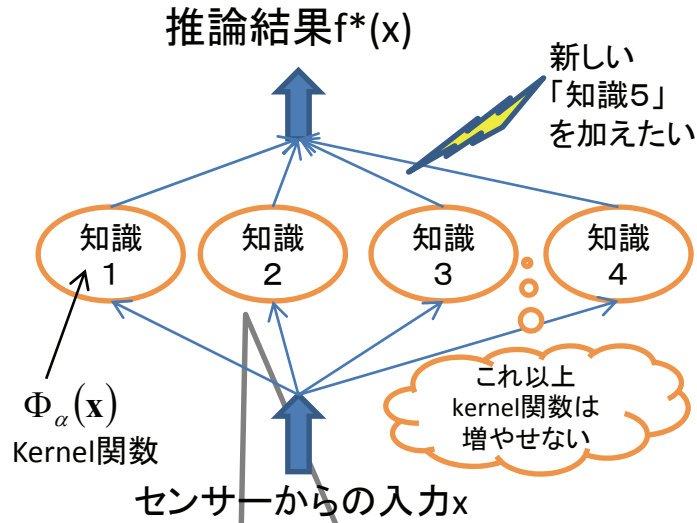
学習機能付き高速  
MPPTコンバータ

その他:

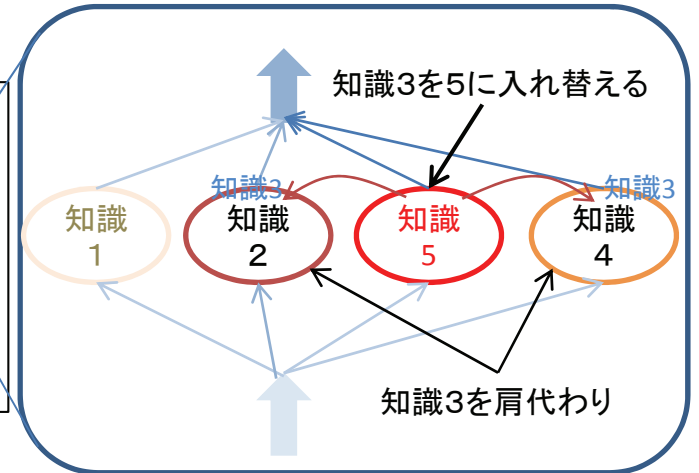
- 学習機能付きPID制御インバータ
- スマートセンサー

# Limited General Regression Neural Network (LGRNN)

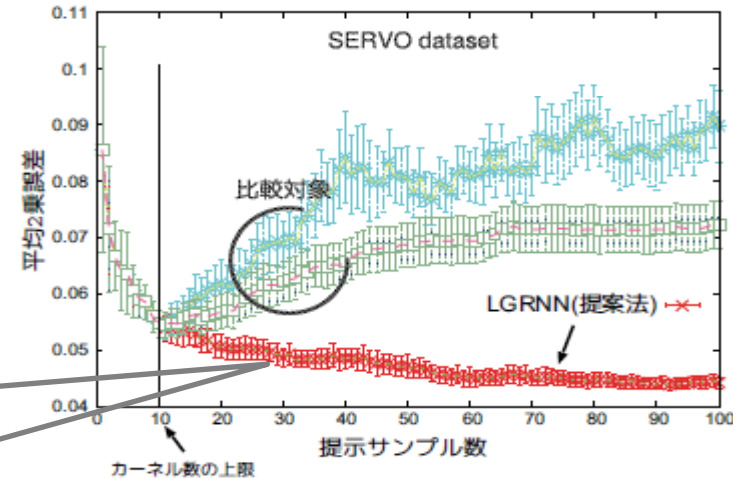
kernelを増やせない場合、次の中から最も予測誤差が小さくなるものを選択



1. もっとも冗長なkernel関数を削除して新kernelを加える。
2. 冗長kernelを似た知識を持つ別のkernel関数に少しずつ肩代わりさせて1を実行。
3. 既存のkernel関数少し変更して新しい知識を学習。
4. 学習しない。(無視する)



ベンチマークテスト結果の例



General Regression Neural Network (Donald 1991)をkernelマシンとして解釈し直す

$$f^*(x) = \frac{\sum_{\alpha=1}^N c_\alpha \Phi_\alpha(x)}{\sum_{\beta=1}^N \Phi_\beta(x)}$$

$$\Phi_\alpha(x) = \exp\left(-\frac{\|x - u_\alpha\|^2}{2\sigma^2}\right)$$

第i Kernel関数の冗長度は以下で計算:

$$\delta_i = \min_{\mathbf{a}_i} \left\{ 1 - 2 \sum_{j \neq i} a_j \Phi_j(\mathbf{u}_i) + \sum_{j \neq i} \sum_{k \neq i} a_j a_k \Phi_j(\mathbf{u}_k) \right\}$$

小さいほど冗長である

LGRNNの場合、提示サンプル数がKernel関数の個数の上限を超えても安定して誤差が下がっている。