Processamento Temporal

Processamento Temporal usando Redes Feedforward

- Tempo
  - Constitui um ingrediente **essencial** no processo de aprendizado
  - É um entidade ordenada que é útil para muitos problemas de reconhecimento de padrões encontrados na prática tais como, **visão, fala, processamento de sinal, e controle de motores**.
  - Através da incorporação do tempo na operação da rede neural, pode-se obter modelos capazes de seguir **variações estatísticas** num processo não estacionário tais como sinal de fala, sinal de radar, flutuações no preço no mercado de ações, etc

Questão
Como nos podermos embutir o tempo na operação de uma rede neural ?

Formas de Representação

- Representação implícita
  - A estrutura temporal do sistema é inserida na estrutura (estática) do modelo de mapeamento (feedforward: MLP, RBF)
  - Está concentrada em memória de linha de atraso
  - **Simples implementação e utilização em ambientes estacionários**.
  - Neste tipo de representação o tempo não é incluído diretamente no modelo, mas sim de uma forma indireta.
  - Desta forma, a rede neural estática é suprida com propriedades dinâmicas, tornando-a sensível quanto a estrutura temporal dos sinais portadores de informação (HAYKIN, 2009).

- Representação Explicita
  - O tempo é representado pelo seu efeito sobre o processamento (dinâmica)
  - Neste caso, são utilizados redes neurais recorrentes

Memória

- O papel primário da memória é transformar uma rede estática em uma rede dinâmica.
- Quando um padrão de atividade particular é aprendido, ele é armazenado no cérebro, de onde pode ser recuperado mais tarde, quando exigido.
- A memória de divide em memória de curto prazo e longo prazo, dependendo do tempo de retenção

- Memória de Longo Prazo
  - É inserida numa rede neural através de algum método de aprendizagem supervisionada
  - Assim, como o conteúdo de informação do conjunto de dados de treinamento é armazenado, total ou parcialmente, nos pesos sinápticos da rede.
  - Se refere ao conhecimento armazenado por um longo período ou permanentemente

- Memória de Curto Prazo
  - Se refere a uma compilação de conhecimento que representa o estado corrente do ambiente. Quaisquer discrepâncias entre o conhecimento armazenado na memória de curto prazo e um novo estado são usadas para atualizar a memória de curto prazo.
  - Entretanto, se a tarefa considerada possuir uma dimensão temporal, é necessário que alguma forma de memória de curto prazo seja inserida, tornando o modelo dinâmico.
  - Uma forma simples de inserir memória de curto prazo na estrutura de uma rede neural é através de atrasos no tempo (*lags), que podem ser* implementados a nível sináptico, ou seja, dentro de uma das camadas ocultas da rede, ou na camada de entrada da rede.

Processamento Temporal

- São problemas que se caracterizam por ser dependente da variável tempo
  - Mercado de ações
  - Mercado de commodities
  - Evolução das vendas de um produto
  - Histórico do consumo de energia elétrica
  - Etc.

Modelagem de sistemas

- As tarefas que estudamos, classificação e regressão, podem ser vistas como atividades de modelagem de sistemas.

- As técnicas de modelagem dependem do tipo de sistema de interesse.

- Um sistema cujo comportamento é constante no tempo, dependente apenas dos valores da entrada, é dito *estático*.

- Neste caso, o conhecimento das entradas instantâneas do sistema determina completamente a sua resposta.

- Um sistema cuja resposta varia no tempo, mesmo que os valores de suas entradas não sejam alterados, é dito *dinâmico*.

- Neste caso, a sua resposta depende não só dos valores instantâneos das suas entradas, mas também dos valores passados, tanto das suas entradas, como das suas saídas.

- Usualmente, o comportamento de um sistema dinâmico é representado pelo seu (vetor de) *estado, que agrega as variáveis do sistema que controlam o seu* comportamento dinâmico.

Redes Neurais Recorrentes

- São redes neurais com um ou mais laços de realimentação

- Dado um perceptron multicamada como bloco construtivo, a aplicação da realimentação pode ser realizada de várias formas
  - Realimentação da saída dos neurônios do perceptron multicamadas para camada de entrada
  - Realimentação dos neurônios da camada escondida para a camada de entrada

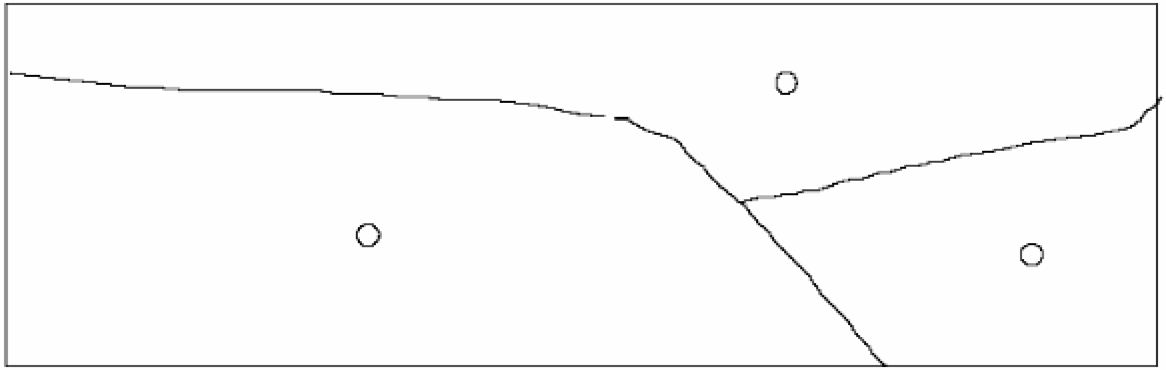
- Quando o perceptron multicamadas tem duas ou mais camadas, as formas possíveis de realimentação expandem ainda mais.

- Basicamente, há dois usos funcionais de redes recorrentes
  - Memórias associativas
  - Mapeamento de entrada-saída

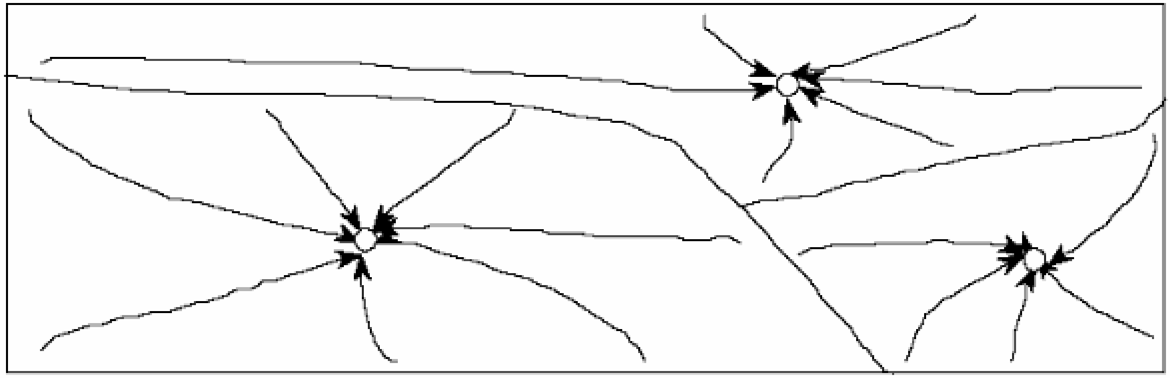
Redes Associativas
Redes de Hopfield

Rede Hopfield

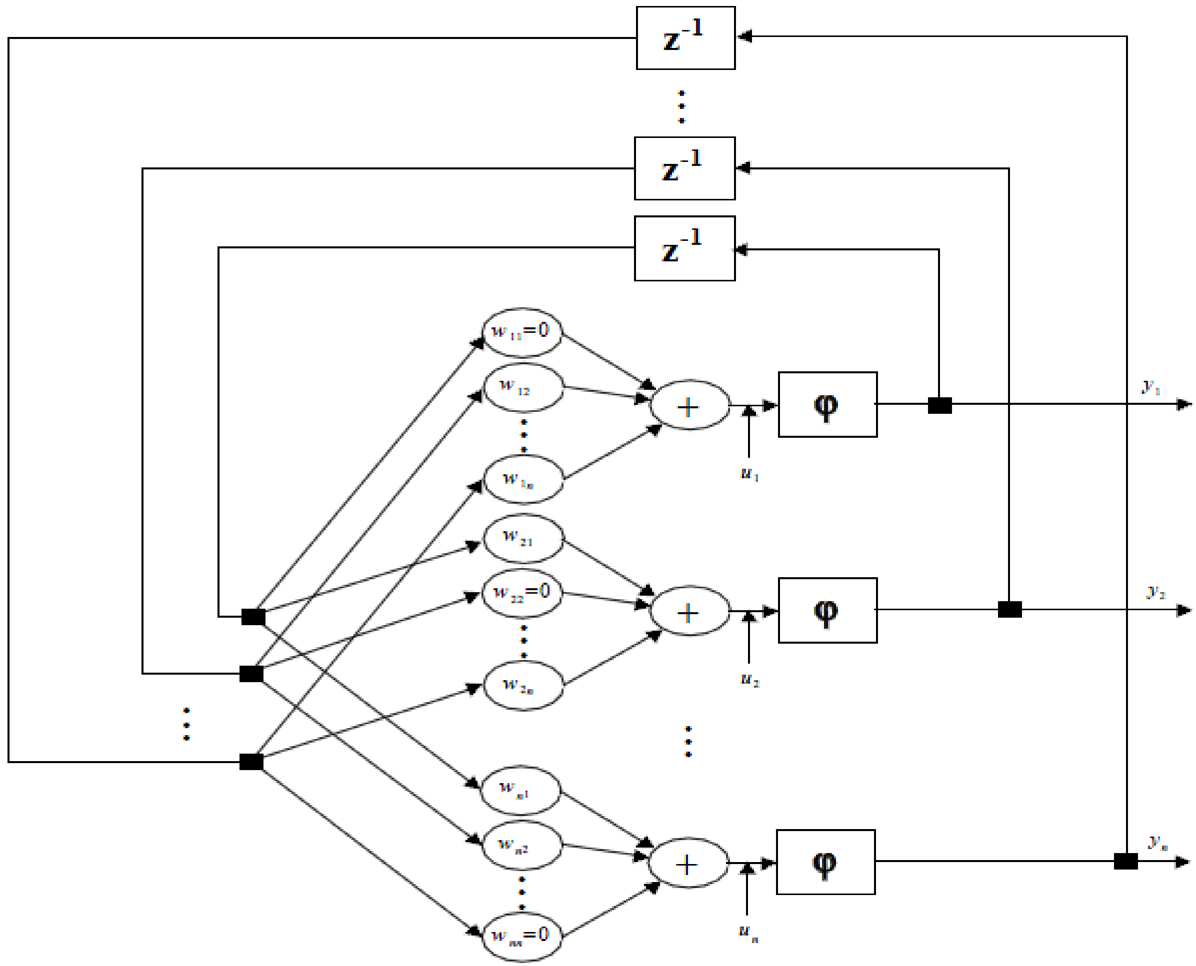
- A rede Hopfield funciona como um mapa que divide um certo espaço de dados *N* em *m* partes



- Assim, dado um certo espaço de dados e um conjunto de pontos *P*, *a rede Hopfield ajustará* seus pesos de forma a possibilitar que todos os pontos no espaço de dados convirjam para os indicados.



Rede Hopfield

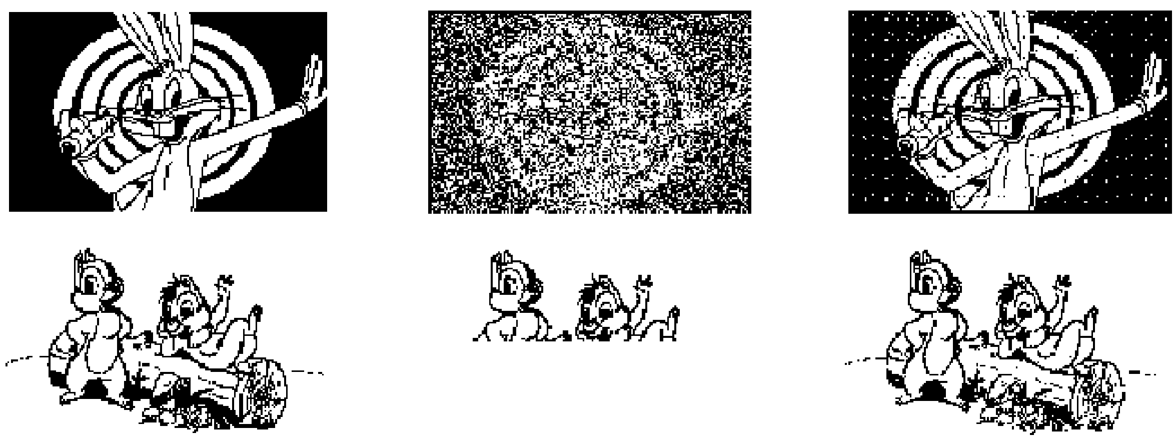


Rede Neural de Hopfield: ênfase no processamento dinâmico

Rede de Hopfield

- A rede de Hopfield é um caso particular de rede recorrente, em que o espaço de estados é discreto

- Como veremos a seguir, ela pode ser vista como uma memória associativa não-linear, ou uma memória endereçável por conteúdo, cuja principal função é restaurar um padrão binário armazenado (item de memória), em resposta à apresentação de uma versão incompleta (papel restaurador) ou ruidosa (papel corretor de erro) deste padrão



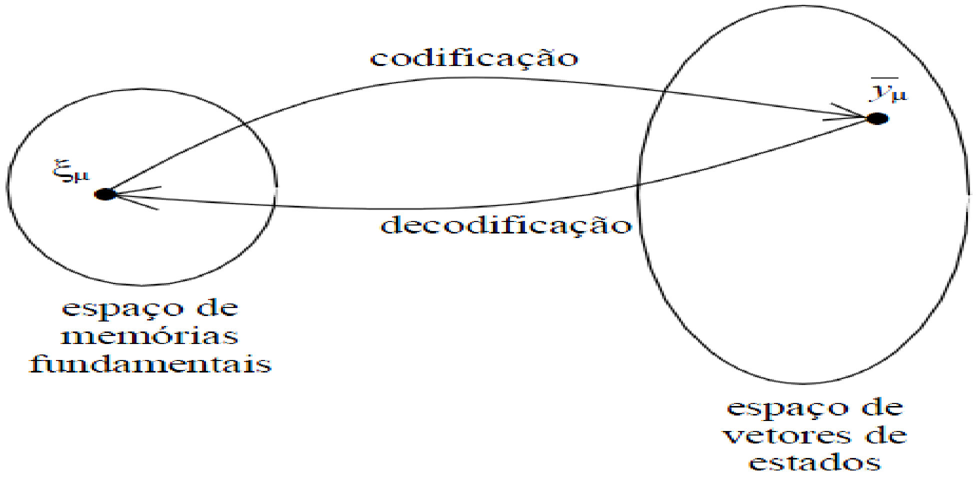
memórias | entradas | padrões restaurados

como memorizar? | como restaurar?

Rede de Hopfield

- Portanto, a recuperação do padrão armazenado na memória se dá a partir de um subconjunto das informações contidas no padrão.

- a essência da memória endereçável por conteúdo é mapear uma memória fundamental ξ_μ em um ponto fixo estável $\bar{y}_u$ do *sistema dinâmico representado* pela rede recorrente.



- logo, a rede neural de Hopfield é um sistema dinâmico não-linear cujo espaço de estados contém um conjunto de pontos fixos estáveis que representam as memórias fundamentais do sistema.

Características operacionais da rede de Hopfield

- o modelo de rede neural de Hopfield utiliza como unidade de processamento básica o neurônio de MCCULLOCH & PITTS (1943) → estado discreto

- *a ativação interna de cada neurônio j (j = 1,…,N) é dada por*

$$u_j = \sum_{\substack{i=1 \\ i \neq j}}^{N} w_{ij} y_j + \theta_j$$

- onde θ_j *é um limiar fixo aplicado externamente ao neurônio j. Tomando, agora,* y_j como o estado do neurônio *j (j = 1,…,N), este será modificado de acordo com a* função de ativação dada pela função sinal:

Características operacionais da rede de Hopfield

$$y_j = sign(u_j) = \begin{cases} +1 & se \quad u_j > 0 \\ -1 & se \quad u_j < 0 \end{cases}$$

- por convenção, se $u_j = 0$, *o estado* y_j *do neurônio permanece em seu valor anterior*, seja ele -1 ou + 1.
- assim, um ponto fixo estável $\bar{y}_u$ *da rede de Hopfield é um estado de convergência* a partir de uma condição inicial que pertence à sua base de atração.
- portanto, estamos considerando o caso $t \rightarrow \infty$ e *estamos fazendo a inclinação da* função sigmoidal $\varphi_j(.)$ *tender a infinito, para reproduzir a função sinal*
- a partir deste ponto, vamos apresentar as duas fases de implementação da rede de Hopfield como memória associativa (endereçável por conteúdo).

- A **primeira fase** corresponde à síntese da dinâmica não-linear com base na definição dos pesos da rede de Hopfield (<u>memorização</u>).

- A **segunda fase** corresponde à restauração de uma dentre as memórias armazenadas (<u>estados de equilíbrio</u>), a partir de um padrão de entrada (<u>estado inicial</u>).

Fase 1: Armazenagem de padrões (memórias fundamentais)

- suponha que se queira armazenar um conjunto de *p padrões dados por vetores N dimensionais* (palavras binárias), denotados por $\{\xi_\mu \mid \mu=1,\ldots,p\}$.

- para tanto, basta definir os pesos pela aplicação da regra de Hebb generalizada, ou regra do produto externo. Seja $\xi_{\mu i}$ *o i-ésimo elemento do vetor* ξ_μ*, então o peso* sináptico conectando o neurônio *i ao neurônio j é definido por*

$$w_{ij} = \frac{1}{N}\sum_{\mu=1}^{p}\xi_{\mu j}\xi_{\mu i}$$

- *sendo que novamente toma-se* $w_{jj}=0$ *(j = 1,...,N)*

- seja *W a matriz* $N\times N$ *de pesos sinápticos, onde* w_{ji} *é o elemento da j-ésima linha e i-ésima coluna. Então, é possível expressar a regra do produto externo na forma:*

$$W = \frac{1}{N}\sum_{\mu=1}^{p}\xi_\mu\xi_\mu^T - \frac{p}{N}I$$

- observe que $W = W^T$, *ou seja,* $w_{ji} = w_{ij}$ *(i,j = 1,…,N)*

Exemplo

- Projetar uma memória associativa recorrente, segundo o modelo de Hopfield, que armazene os seguintes 2 padrões tridimensionais, como memórias fundamentais:

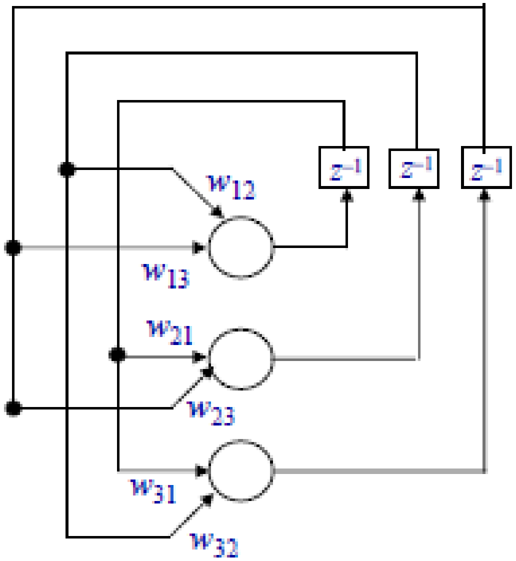
$$\xi_1 = [\,+1\ -1\ +1\,]^T \text{ e } \xi_2 = [-1\ +1\ -1]^T$$

- Calcular os pesos pela expressão:

$$W = \frac{1}{N}\sum_{\mu=1}^{p} \xi_\mu \xi_\mu^T - \frac{p}{N} I$$

- A partir de um vetor de prova $\mathbf{x = [\,-1\ \ +1\ \ +1\,]}^T$ realize atualizações assíncronas do estado da rede até recuperar a memória fundamental correspondente.

Exemplo



$$\mathbf{w} = \frac{1}{N}\sum_{\mu=1}^{p} \xi_\mu \xi_\mu^T - \frac{p}{N}\mathbf{I}$$

Com: $\xi_1 = [+1\ -1\ +1]^T$ e $\xi_2 = [-1\ +1\ -1]^T$

Da definição do termo da matriz identidade :

$$\frac{p}{N}\mathbf{I} = \frac{2}{3}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\frac{1}{N}\sum_{\mu=1}^{p} \xi_\mu \xi_\mu^T = \frac{1}{3}\begin{bmatrix} +1 \\ -1 \\ +1 \end{bmatrix} [+1\ -1\ +1] + \frac{1}{3}\begin{bmatrix} -1 \\ +1 \\ -1 \end{bmatrix} [-1\ +1\ -1] = \frac{1}{3}\begin{bmatrix} +2 & -2 & +2 \\ -2 & +2 & -2 \\ +2 & -2 & +2 \end{bmatrix}$$

Então : $$\mathbf{w} = \frac{1}{3}\begin{bmatrix} 0 & -2 & +2 \\ -2 & 0 & -2 \\ +2 & -2 & 0 \end{bmatrix}$$

Fase 2: Recuperação dos padrões (estados de equilíbrio estáveis)

- durante a fase de recuperação dos padrões armazenados (memórias), um vetor *N dimensional y é tomado como o estado inicial (condição inicial) da rede de* Hopfield. Obviamente, os elementos de *y assumem valores –1 ou +1.*
- geralmente, este vetor *y vai representar uma versão incompleta ou ruidosa da* memória fundamental que foi armazenada na rede.
- o processo de recuperação da memória armazenada obedece a uma regra dinâmica denominada ajuste assíncrono. Um único neurônio *j da rede é escolhido* aleatoriamente para ter sua saída y_j *(que agora está associada ao estado da rede)* recalculada em função do valor de u_j
- assim, o ajuste do estado da rede, de uma iteração para outra, é determinístico, mas a escolha do neurônio cujo estado será atualizado é aleatória.
- este ajuste assíncrono prossegue até que não haja mais mudanças de estado a processar, ou seja, até que a rede atinja um ponto de equilíbrio caracterizado por:

Fase 2: Recuperação dos padrões (estados de equilíbrio estáveis)

$$\bar{y}_j = \operatorname{sgn}\left(\sum_{i=1}^{N} w_{ij} y_i + \theta_j \right), \quad j = 1, \cdots, N$$

- ou em notação matricial:

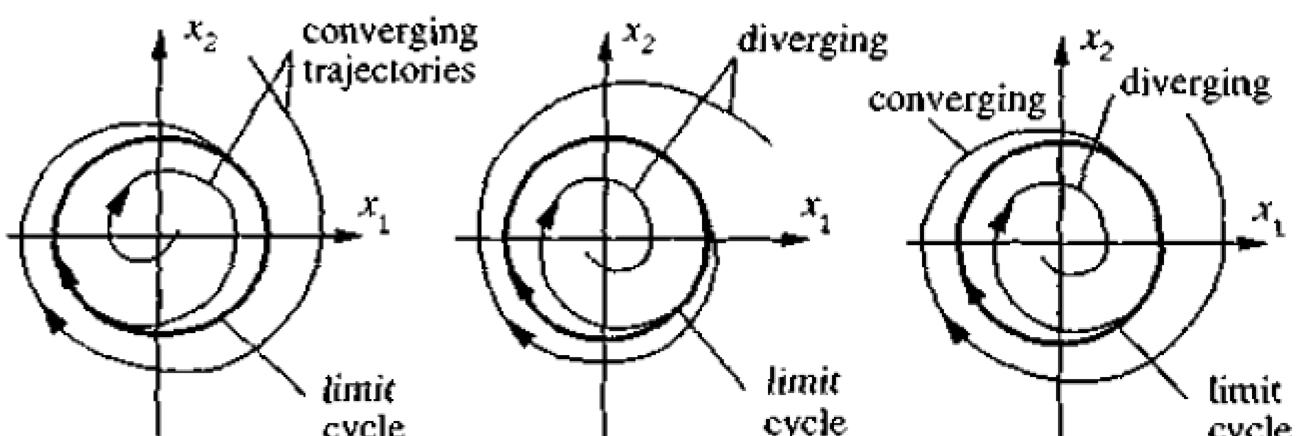
$$\bar{\mathbf{y}} = \operatorname{sgn}(\mathbf{W}\bar{\mathbf{y}} + \boldsymbol{\theta})$$

- a rede de Hopfield sempre vai convergir para um ponto de equilíbrio se o ajuste for assíncrono.

- Se o ajuste fosse síncrono, ou seja, todos os neurônios tendo seus estados recalculados em paralelo, ciclos limites de até 2 pontos poderiam ser produzidos (BRUCK, 1990).

Adendo – Ciclo Limite

- No plano de fase, ciclo-limite é uma curva fechada (caracterizando a oscilação) e isolada
- Três tipos de ciclos-limite, dependendo da trajetória em sua vizinhança
  - Ciclo-limite estável: as trajetórias na vizinhança convergem para ele à medida que o tempo evolui
  - Ciclo-limite instável: as trajetórias na vizinhança divergem dele à medida que o tempo evolui
  - Ciclo-limite semi-estável: algumas trajetórias na vizinhança dele convergem, e outras divergem



Recapitulação

- não-linearidade é condição necessária para produzir múltiplos atratores no espaço de estados de sistemas dinâmicos.

- Hopfield resolveu parcialmente o seguinte problema: *dado um conjunto de estados específicos que devem estar associados a memórias fundamentais, como gerar um sistema dinâmico não-linear que apresente pontos de equilíbrio estável justamente nestes estados específicos?*

- se este sistema dinâmico não-linear puder ser sintetizado, então vai existir uma superfície de energia com mínimos locais nos referidos estados específicos, sendo que a dinâmica do sistema vai atuar no sentido de conduzir o estado inicial do sistema a um dos mínimos locais da superfície de energia (particularmente àquele em cuja base de atração se encontra a condição inicial).

A emergência de memória associativa

- quando um estado de equilíbrio estável (ponto de mínimo local da função de energia) é atingido, não há como reduzir ainda mais a energia, fazendo com que o estado da rede fique <u>invariante frente à dinâmica</u>.

- assim, para garantir a emergência de memória associativa, duas condições devem ser satisfeitas:
  - 1. as memórias fundamentais devem ser armazenadas como estados estáveis da rede;
  - 2. estes estados estáveis correspondentes às memórias fundamentais devem ter uma base de atração de dimensão não-nula.

Atratores espúrios

- quando a rede neural de Hopfield armazena *K memórias fundamentais através do* ajuste de seus pesos pela regra de Hebb generalizada, os estados estáveis presentes na superfície de energia não vão se restringir aos estados associados às memórias fundamentais armazenadas. Todos os estados estáveis **não** associados às memórias fundamentais armazenadas são denominados atratores espúrios.
- os atratores espúrios existem em virtude dos seguintes fatores:
  1. a função de energia *E é simétrica, no sentido de que os estados correspondentes ao* reverso das memórias fundamentais armazenadas também são estados estáveis;
  2. toda combinação linear de um número ímpar de estados estáveis também vai ser um estado estável (AMIT, 1989).
  3. para um grande número *K de memórias fundamentais, a função de energia vai* produzir **pontos de equilíbrio que não estão** correlacionados com nenhuma das memórias fundamentais armazenadas na rede (inflexibilidade da superfície de energia).

Capacidade de memória da rede de Hopfield

- infelizmente, as memórias fundamentais utilizadas para gerar os pesos da rede de Hopfield, de acordo com a seguinte equação:

$$w_{ij} = \frac{1}{N}\sum_{\mu=1}^{p} \xi_{\mu j}\xi_{\mu i}$$

nem sempre conduzem a estados estáveis.

- desse modo, a possível existência de estados espúrios, aliada à possibilidade de instabilidade das memórias fundamentais, tendem a reduzir a eficiência da rede de Hopfield como uma memória endereçável por conteúdo.
- considere a ativação interna do neurônio *j, dada na forma:*

$$u_j = \sum_{i=1}^{N} w_{ji} y_i$$

- onde, para efeito de generalidade, estamos assumindo agora que $w_{jj} \neq 0$.

Capacidade de memória da rede de Hopfield

- denominando x um estado genérico da rede, temos que

$$u_j = \frac{1}{N}\sum_{i=1}^{N}\sum_{\mu=1}^{p}\xi_{\mu j}\xi_{\mu i}x_i = \frac{1}{N}\sum_{\mu=1}^{p}\xi_{\mu j}\sum_{i=1}^{N}\xi_{\mu i}x_i$$

- Considere agora o caso especial em que o estado genérico x é tomado como uma das memórias fundamentais armazenadas na rede, por exemplo, ξ_v

$$u_j = \frac{1}{N}\sum_{\mu=1}^{p}\xi_{\mu j}\sum_{i=1}^{N}\xi_{\mu i}\xi_{vi} \Rightarrow u_j = \xi_{vj} + \frac{1}{N}\sum_{\substack{\mu=1\\ \mu\neq v}}^{p}\xi_{\mu j}\sum_{i=1}^{N}\xi_{\mu i}\xi_{vi}$$

- A parcela mais à esquerda, ξ_v, é simplesmente o j-ésimo elemento da memória fundamental ξ_{vj}, constituindo o valor desejado (sinal) para μ_j, já que a memória fundamental deve ser um estado estável. Este resultado justifica a necessidade a necessidade da divisão por N na geração dos pesos

- a parcela mais à direita, $\frac{1}{N}\sum_{\substack{\mu=1\\ \mu\neq v}}^{p}\xi_{\mu j}\sum_{i=1}^{N}\xi_{\mu i}\xi_{vi}$, é o ruído existente quando os padrões não são ortogonais, ou seja, quando $\sum_{i=1}^{N}\xi_{\mu i}\xi_{vi} \neq 0$

Capacidade de memória da rede de Hopfield

- através de um estudo estatístico, assumindo, dentre outros aspectos, que as memórias fundamentais são formadas por padrões gerados aleatoriamente, é possível mostrar que a relação sinal-ruído é dada aproximadamente por

$$\rho \cong \frac{N}{K}, \textit{ para valores elevados de } K$$

- com isso, o componente de memória fundamental ξ_v será estável (em sentido probabilístico) se, e somente se, a relação sinal-ruído for suficientemente alta.
- valores sugeridos na literatura para ρ:

$$\rho = 7.25, ou \; \frac{1}{\rho} = 0{,}138 \; (K = 138 \quad quando \quad N = 1000)$$

$$\rho \geq 2 \ln N \Rightarrow K \leq \frac{N}{2 \ln N} (K \leq 72 \;\; quando \;\; N = 1000)$$

Extensões (Parte I)

- a rede neural de Hopfield apresentada anteriormente usa dinâmica discreta e toma o neurônio de McCulloch-Pitts como unidade básica. No entanto, um desempenho muito superior, quanto à capacidade de memória, pode ser obtido empregando funções de ativação contínuas e não-monotônicas (MORITA, 1993). Obviamente, como a função de ativação passa a assumir valores em um intervalo, é necessário aplicar a função sinal para recuperar a memória a partir da saída estabilizada da rede.

- o resultado de MORITA (1993) trás como consenqüência:
  1. aumento da capacidade de memória de $\frac{N}{2\ln N}$ para 0.4N, *onde N é o número de* neurônios;
  2. desaparecimento dos estados espúrios.

Extensões (Parte II)

- uma regra mais eficiente para definição dos pesos da rede de Hopfield convencional, em substituição à regra de Hebb generalizada, é a regra de projeção.
- ela tem a desvantagem de não apresentar uma motivação biológica, mas a vantagem de explorar as propriedades algébricas dos pontos de equilíbrio.
- ξ_μ será um ponto de equilíbrio se $\mathbf{W}\xi_\mu = \xi_\mu$, $\mu=1,\ldots, K$
- Seja $P=[\xi_1 ,\ldots, \xi_K]$, então temos que $\mathbf{WP} = \mathbf{P}$
- Uma solução para W é dada na forma: $\boxed{\mathbf{W = P(P^T P)^{-1}P^T}}$
- Para que exista $\mathbf{(P^T P)^{-1}}$, basta que os vetores ξ_μ, $\mu=1,\ldots, K$ sejam linearmente independentes, pois esta condição garante que a matriz P tem posto completo.
- A matriz $P^+=\mathbf{(P^T P)^{-1}P^T}$ é denominada pseudo-inversa de **P** (Moore-Penrose)
- $\mathbf{W = PP^+}$ é a matriz de projeção ortogonal de vetores do $\Re^N$ para o subespaço cuja base é o conjunto de vetores ξ_μ, $\mu=1,\ldots, K$

Exercício

- Seja $P=[\xi_1, \ldots, \xi_K]$, então temos que $\mathbf{WP} = \mathbf{P}$

- Prove que uma solução para W é dada na forma:

$$\mathbf{W} = \mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T$$

Mapeamento de Entrada-Sáida

Redes neurais recorrentes de tempo discreto

- Redes neurais recorrentes são estruturas de processamento capazes de representar uma grande variedade de comportamentos dinâmicos.

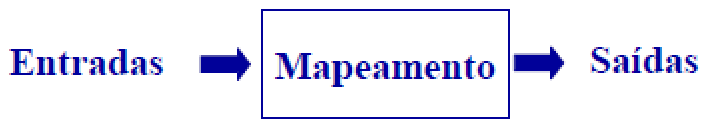
- A presença de realimentação de informação permite a criação de representações internas e dispositivos de memória capazes de processar e armazenar informações temporais e sinais sequenciais.

- A presença de conexões recorrentes ou realimentação de informação pode conduzir a comportamentos complexos, mesmo com um número reduzido de parâmetros.

- Como estruturas de processamento de sinais, redes neurais recorrentes se assemelham a filtros não-lineares com resposta ao impulso de duração infinita (NERRAND *et al., 1993)*.

Tarefas dinâmicas

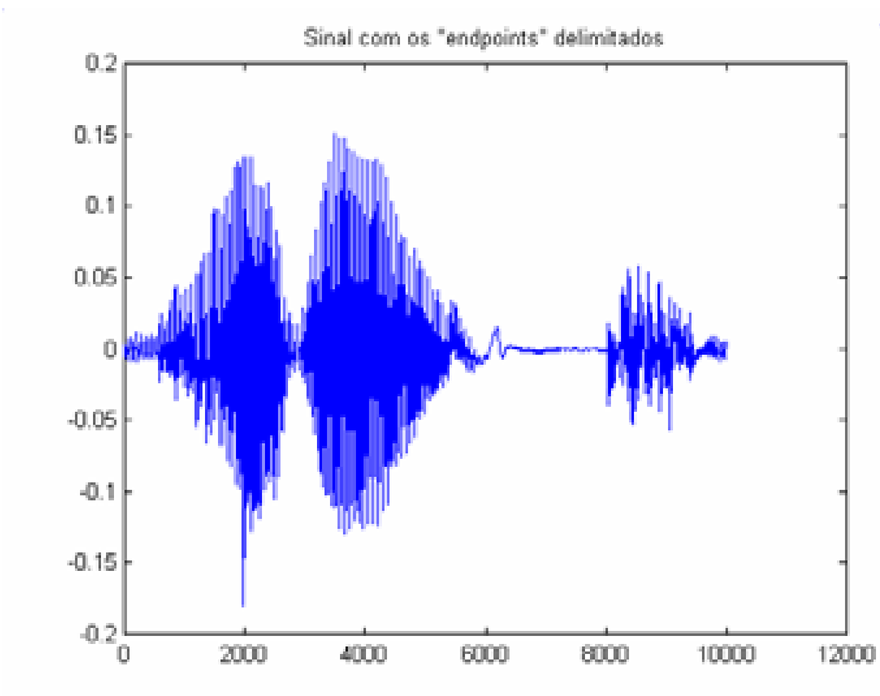
- Geração de modelo preditivo
  - Mapeamento de entrada-saída dinâmico



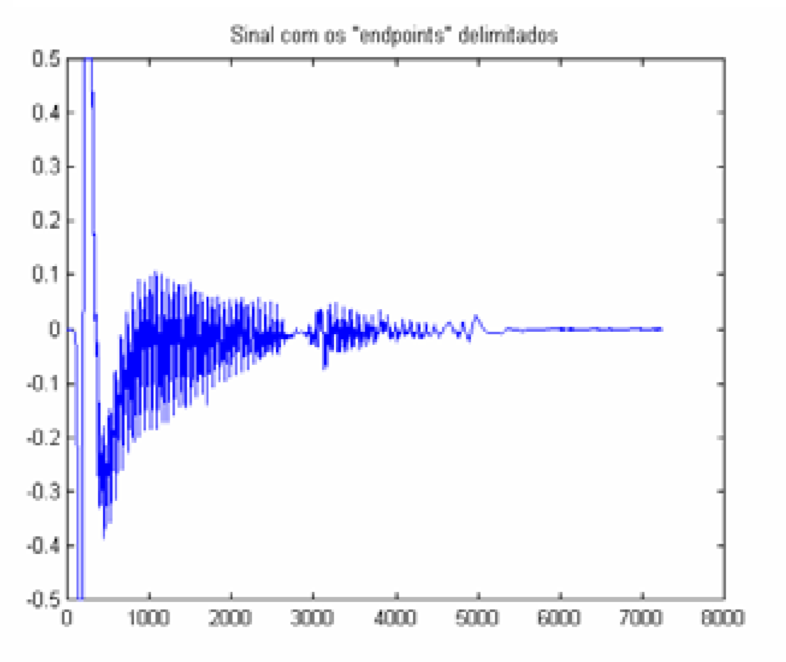
  - **Classificação**: saídas discretas representam rótulos de classe.
  - **Regressão**: saídas contínuas representam valores de variáveis dependentes: previsão de valores futuros
  - Controlador de um sistema dinâmico

Classificação: Reconhecimento de palavras faladas

- Reconhecer palavras de um dicionário conhecido
- Numa situação dinâmica, deve-se detectar o início da palavra



direita



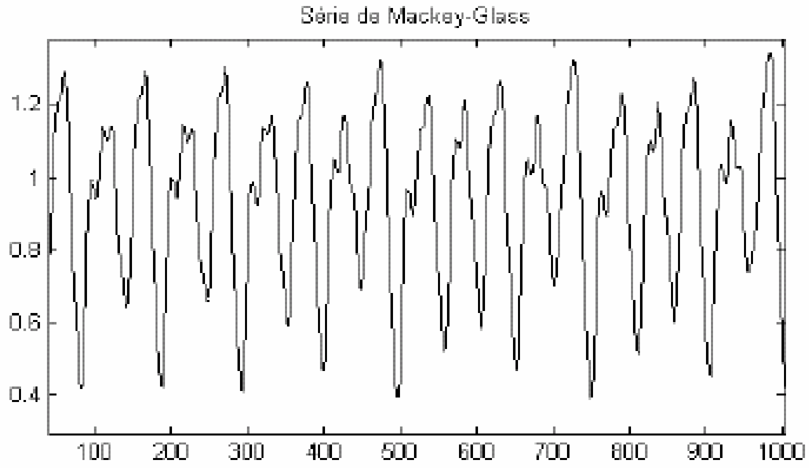
pare

Regressão: Previsão de séries temporais

- Em várias aplicações, é importante usar observações de uma série temporal disponíveis no tempo t, *para prever o seu valor num tempo futuro* $t + l$.

- Por exemplo, as vendas z_t *no mês corrente* e *as vendas* $z_{t-1}, z_{t-2}, z_{t-3}, \ldots$, *nos* meses anteriores, podem ser usadas para prever vendas para tempos $l = 1, 2, \ldots, 12$ meses adiante.

- Na prática, em várias aplicações de interesse, ocorrem séries que apresentam comportamento caótico

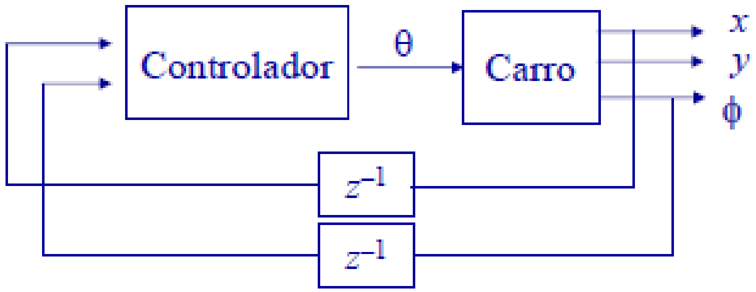


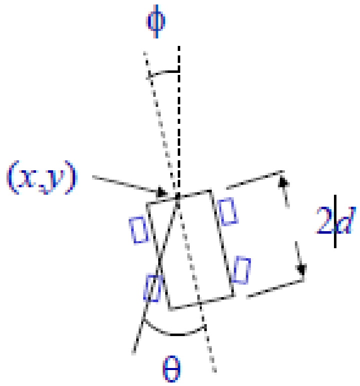
Controle de um sistema dinâmico

- Problema: estacionar um carro de ré em um local especificado

- Variável de controle: ângulo das rodas dianteiras (θ)

- Estado do carro:
  - Posição central da traseira (x,y)
  - Ângulo do carro em relação ao eixo dos y (φ)
  - Estado final: (xf, yf, θf)





Tarefas dinâmicas

- Geração de modelo descritivo

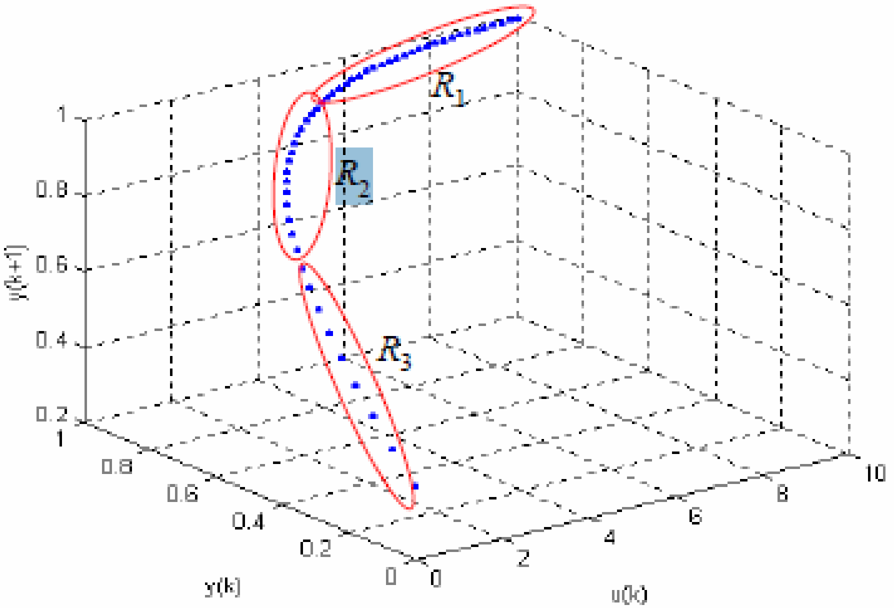
Entradas → Modelo

  - Determinação de uma configuração de centróides de agrupamentos dos dados no tempo.

  - Determinação de um modelo de agrupamento dos dados no tempo.

Identificação de Agrupamentos

- A partir de uma matriz de regressão dinâmica, é determinada uma estrutura de agrupamentos.



Redes Recorrentes

Redes Neurais Recorrentes de Tempo Discreto

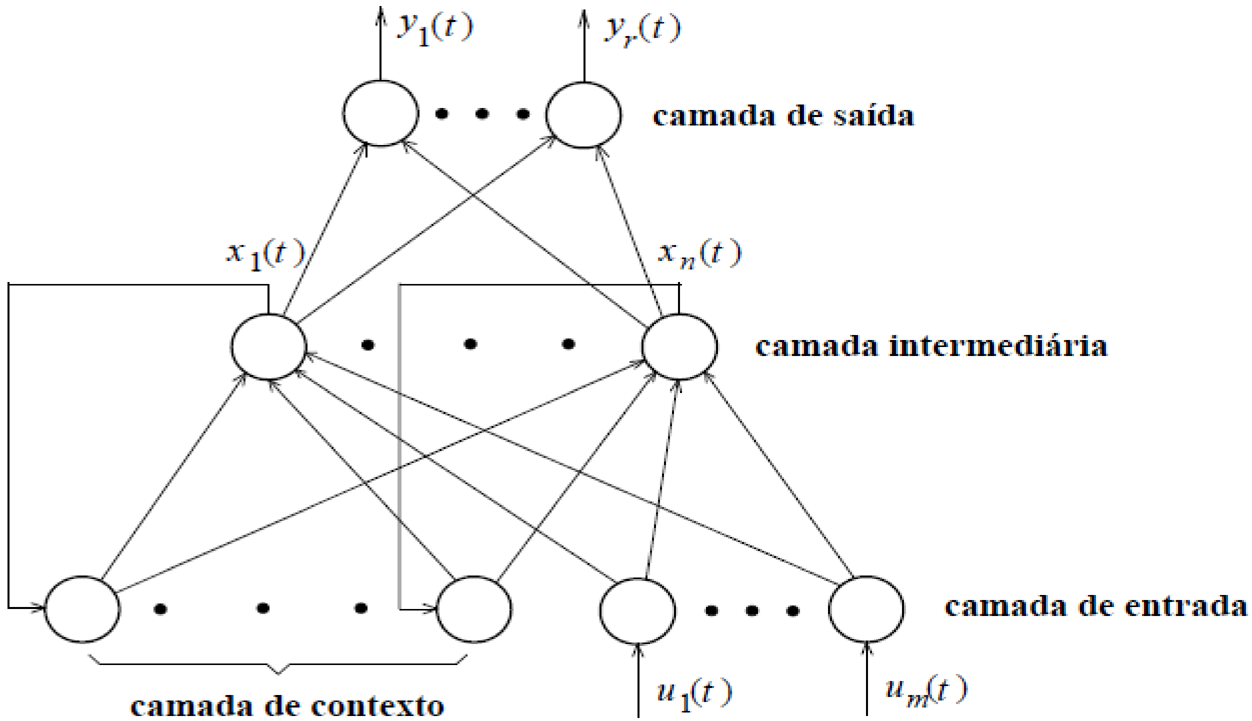
- redes neurais recorrentes são estruturas de processamento capazes de representar uma grande variedade de comportamentos dinâmicos.

- a presença de realimentação de informação permite a criação de representações internas e dispositivos de memória capazes de processar e armazenar informações temporais e sinais sequenciais.

- a presença de conexões recorrentes ou realimentação de informação pode conduzir a comportamentos complexos, mesmo com um número reduzido de parâmetros.

- como estruturas de processamento de sinais, redes neurais recorrentes se assemelham a filtros não-lineares com resposta ao impulso infinita (NERRAND *et al., 1993)*.

Modelagem de sistemas dinâmicos lineares



Estrutura detalhada da Rede de Elman (Elman, 1990)

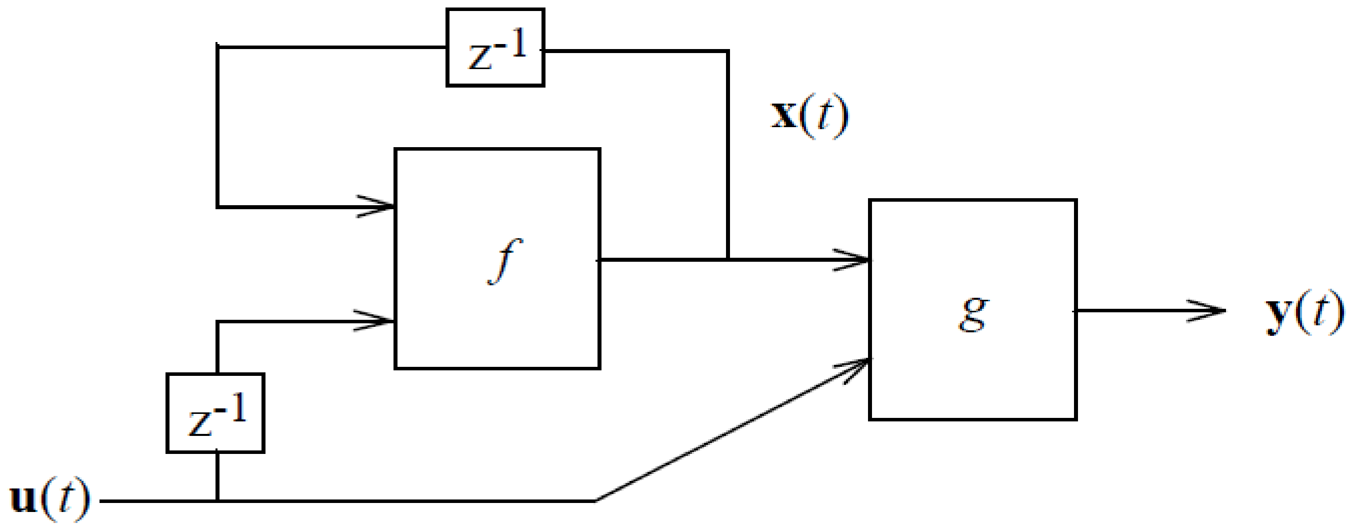
$$z_{in}(t) = C\mathbf{x}(t-1) + A\mathbf{u}(t)$$

$$z(t) = f(z_{in}(t))$$

$$\mathbf{y}(t) = Bx(t)$$

(aproxima qualquer dinâmica linear nesta representação)

Modelagem de sistemas dinâmicos não-lineares



- Representação por espaço de estados de um sistema dinâmico não-linear

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t))$$

$$\mathbf{y}(t) = g(\mathbf{x}(t), \mathbf{u}(t))$$

$$onde\ \ \mathbf{u}(t) \in \Re^m,\ \mathbf{x}(t) \in \Re^n, \mathbf{y}(t) \in \Re^r,\ f : \Re^m \to \Re^n\ \ e\ \ g : \Re^{n \times m} \to \Re^r$$

O efeito da recorrência no processo de adaptação

- redes neurais não-recorrentes podem ser interpretadas como poderosos operadores de transformação de representação, mas não são capazes de reutilizar a informação transformada, produzindo apenas mapeamentos estáticos.

- esta é a razão para que este tipo de rede neural encontre dificuldade em representar comportamento dinâmico, já que o vetor de saída da rede neural, denominado $\hat{y}(t)$ **,** depende apenas do vetor de entrada definido no mesmo instante, denominado $\mathbf{x}(t)$. Isto conduz a mapeamentos do tipo:

$$\hat{\mathbf{y}}(t) = RN(\mathbf{x}(t), \theta(t))$$

- onde $\theta(t)$ *denota o vetor de parâmetros no instante t, sendo de dimensão fixa no caso* de redes neurais semi-paramétricas e de dimensão variável no caso de redes neurais não-paramétricas (por exemplo, redes neurais construtivas).

O efeito da recorrência no processo de adaptação

- por outro lado, em redes neurais recorrentes o tempo é representado pelo seu efeito real no processamento. Considerando apenas a existência de recorrência externa, ou seja, realimentação apenas da informação de saída da rede neural, resultam modelos do tipo:

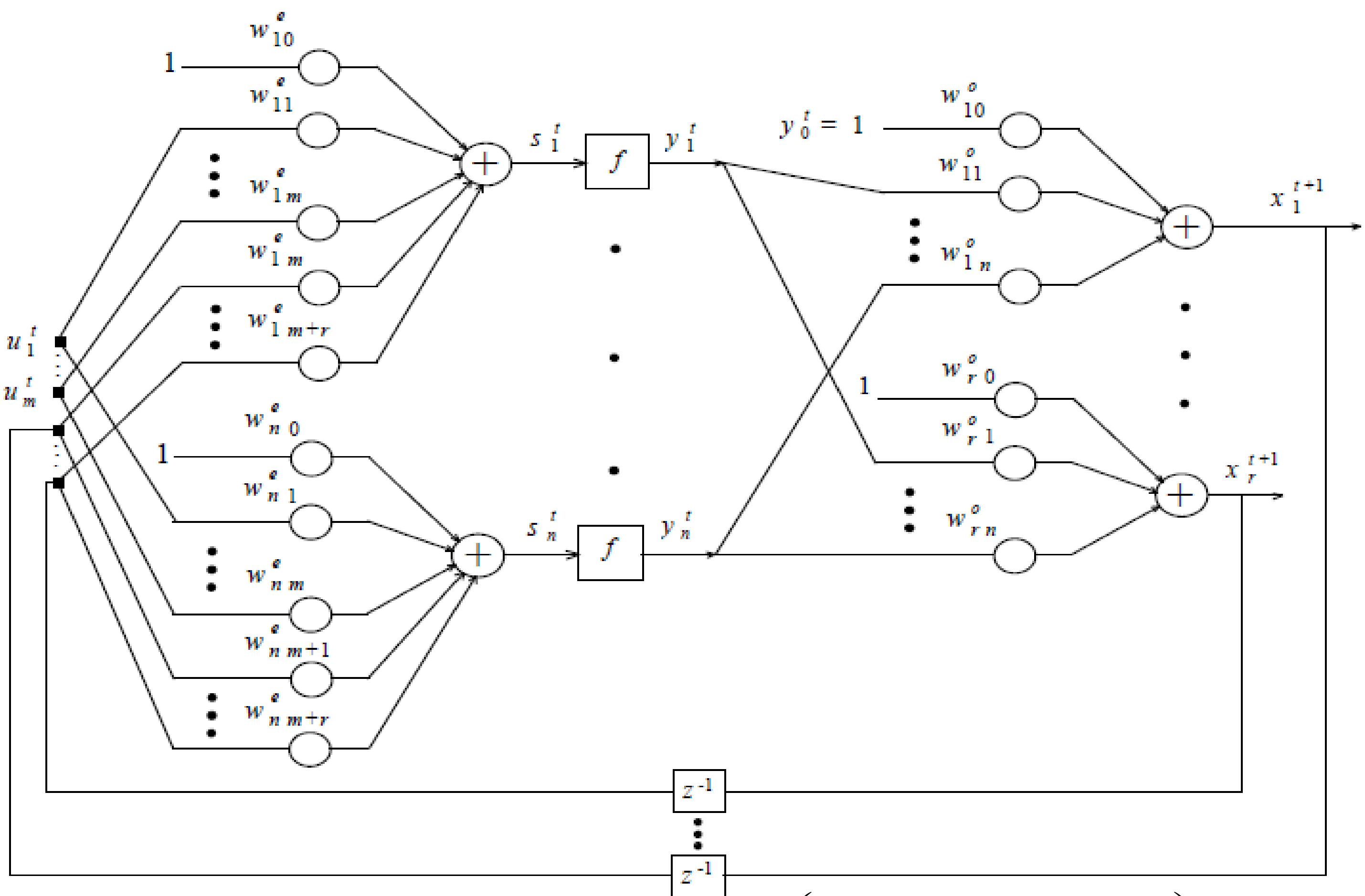
$$\hat{\mathbf{y}}(t) = RN_{rec}(\mathbf{x}(t), \hat{\mathbf{y}}(t-1), \theta(t))$$

- por substituição recursiva desta equação nela mesma, fica claro que o vetor de saída da rede neural depende da história do vetor de entrada e do vetor de parâmetros.

- resulta, então, uma rede neural paramétrica com recorrência externa, apresentada na Figura 3 a seguir.

- repare que o processo de treinamento vai envolver duas dinâmicas acopladas: a dinâmica da rede neural e a dinâmica do ajuste de pesos.

Rede Neural com Recorrência Externa



$$x_k^{t+1} = \sum_{j=0}^{n} w_{kj}^o y_j^t = \sum_{j=1}^{n} w_{kj}^o f(s_j^t) + w_{k0}^o = \sum_{j=1}^{n} w_{kj}^o f\left(\sum_{i=1}^{m} w_{ji}^e u_i^t + \underbrace{\sum_{i=m+1}^{m+r} w_{ji}^e x_{i-m}^t}_{termo\ adicional} \right) + w_{k0}^o$$

Treinamento supervisionado para redes recorrentes

- a disponibilidade de redes neurais recorrentes de importância prática está associada à existência de algoritmos de otimização eficientes para o ajuste de parâmetros.

- observe que uma rede neural do tipo perceptron com uma camada intermediária é um caso particular da rede recorrente apresentada acima.

- caso o vetor de parâmetros seja constante ao longo do tempo, é possível substituir $\theta(t)$ por q nas equações (1) e (2). Com isso, fica claro que, ao contrário do modelo de rede neural não-recorrente, o modelo de rede neural recorrente é uma função composta de θ.

- Logo, a análise variacional dos modelos com e sem recorrência produz os seguintes resultados:

- rede neural não-recorrente: $\hat{\mathbf{y}}(t) = RN(\mathbf{x}(t), \theta)$

$$\frac{\partial \mathbf{y}(t)}{\partial \theta} = \frac{\partial RN}{\partial \theta}$$

Treinamento supervisionado para redes recorrentes

- Rede neural recorrente

$$\hat{\mathbf{y}}(t) = RN_{rec}(\mathbf{x}(t), \hat{\mathbf{y}}(t-1), \theta)$$

$$\frac{\partial \hat{\mathbf{y}}(t)}{\partial \theta} = \frac{\partial RN_{rec}}{\partial \theta} + \underbrace{\frac{\partial RN_{rec}}{\partial \hat{\mathbf{y}}(t-1)} \frac{\partial \hat{\mathbf{y}}(t-1)}{\partial \theta}}_{termo\ adicional}$$

- o problema de modelagem de sistemas dinâmicos de tempo discreto utilizando redes neurais recorrentes semi-paramétricas pode ser colocado na forma:

- $seja\ \{(\mathbf{x}(t), \mathbf{y}(t))\}_{t=1}^{N} \in X \times \Re^{r}$, $onde\ X \subset \Re^{m}$, um conjunto de dados amostrados de um sistema dinâmico de tempo discreto a ser identificado mapeamento realizado pela rede

- $seja\ \ RN_{rec}(\mathbf{x}(t), \hat{\mathbf{y}}(t), \theta) : X \times \Re^{r} \times \Re^{p} \rightarrow \Re^{r}$ o mapeamento realizado pela neural recorrente, onde $\theta \in \Re^{P}$ *é um vetor contendo todos os P parâmetros da rede* neural, ordenados de forma arbitrária, mas fixa;

- resolva o seguinte problema de otimização:

$$\min_{\theta}\ J(\theta) = \min_{\theta}\ \frac{1}{2} \sum_{t=1}^{N} \left\| RNrec(\mathbf{x}(t), \mathbf{y}(t-1), \theta) - \mathbf{y}(t) \right\|^{2}$$

Treinamento supervisionado para redes recorrentes

- a solução deste problema de otimização não-linear só pode ser obtida via processos numéricos iterativos, já que soluções analíticas não estão disponíveis, principalmente devido à natureza não-linear do modelo de aproximação $RN_{rec}(\mathbf{x}(t), \hat{\mathbf{y}}(t-1), \theta)$

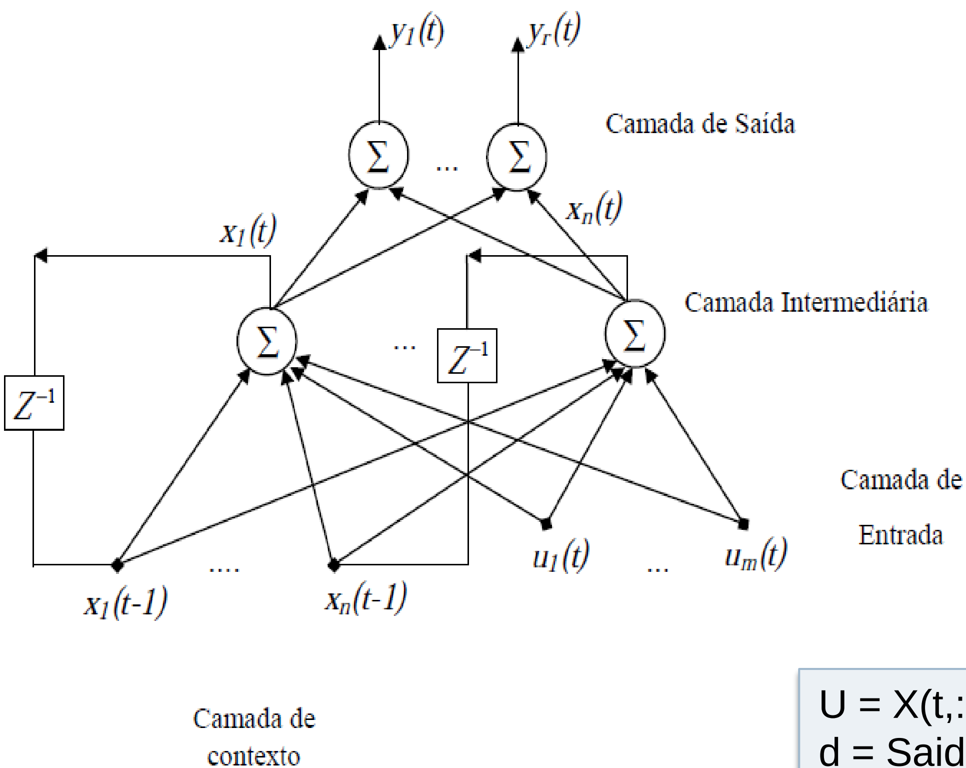
- ·A idéia básica deste processo númerico é realizar ajustes iterativos no vetor de parâmetros $\theta \in \Re^p$ *sempre em direções em que a função-objetivo J*(θ) *decresça.*

- observe que *J*(θ) *define uma superfície no espaço* $\Re^{p+1}$*, ou seja, a dimensão do espaço* de busca é igual ao número de parâmetros a serem ajustados, havendo uma dimensão a mais que indica o valor da função-objetivo para cada θ.

- como o vetor de parâmetros $\theta \in \Re^p$ *não vai ser constante durante o treinamento,* aproximações para o cálculo do vetor gradiente devem ser adotadas.

Excursão ilustrativa por outros tipos de arquiteturas recorrentes

- Estrutura de uma Rede de Elman

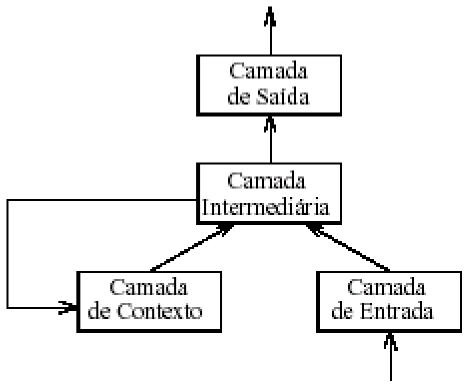



```
U = X(t,:)';        % entrada externa atual
d = Saida(t,:)';  % saídas desejadas
S = A*Zold + B*U;
Z = tanh(S);  % entradas para a camada de saida
Y = C*[Z;1];
```

Caracteristicas

- É uma Rede Neural Temporal com realimentação global interna, onde cada um dos neurônios escondidos tem realimentação para a entrada de todos eles.
- È uma rede parcialmente recorrente, pois as conexões feedforward são modificáveis, e as conexões recorrentes são fixas.
- Dentre as arquiteturas de redes neurais recorrentes, a rede de Elman é a mais simples (Elman, 1990)

Características



- **Camada de entrada**: composta de neurônios que recebem um sinal externo e o propagam sem alterá-lo
- **Camada de saída**: composta por neurônios lineares cujas saída são somas ponderadas de seus respectivos sinais de entrada

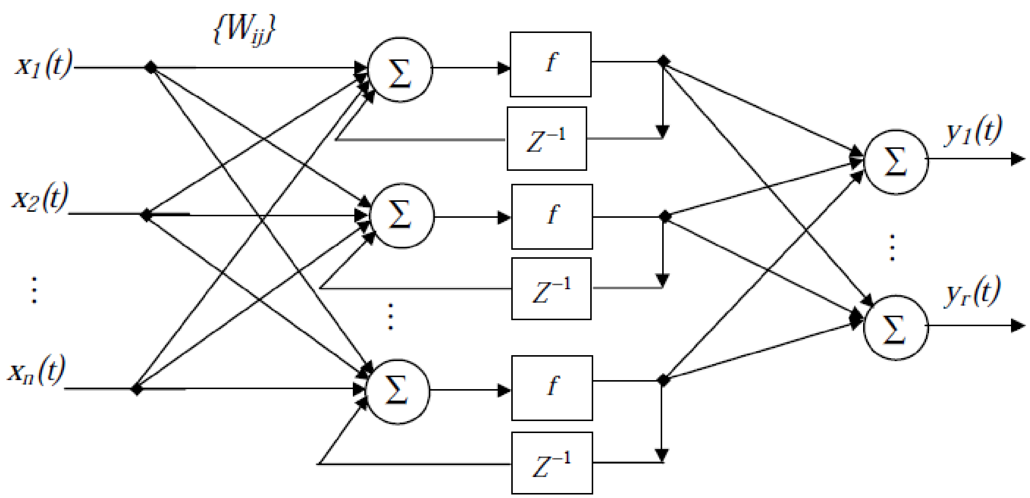
- **Camada intermediária:** composta por neurônios que apresentam funções de ativação ou não-lineares, dependendo do tipo de aplicação.
- Camada de contexto: composta por neurônios que apenas memorizam funções de ativação dos neurônios da camada intermediária, operando como atrasadores de um instante de tempo.

Caracteristicas

- Linearidade na camada de saída
- Ao efetuar a primeira iteração (k=1) é conveniente fixar os valores iniciais das saída dos neurônios escondidos e de saída (k=0).
- Só na primeira iteração a rede atuará como se não fosse recorrente

Redes Recorrentes

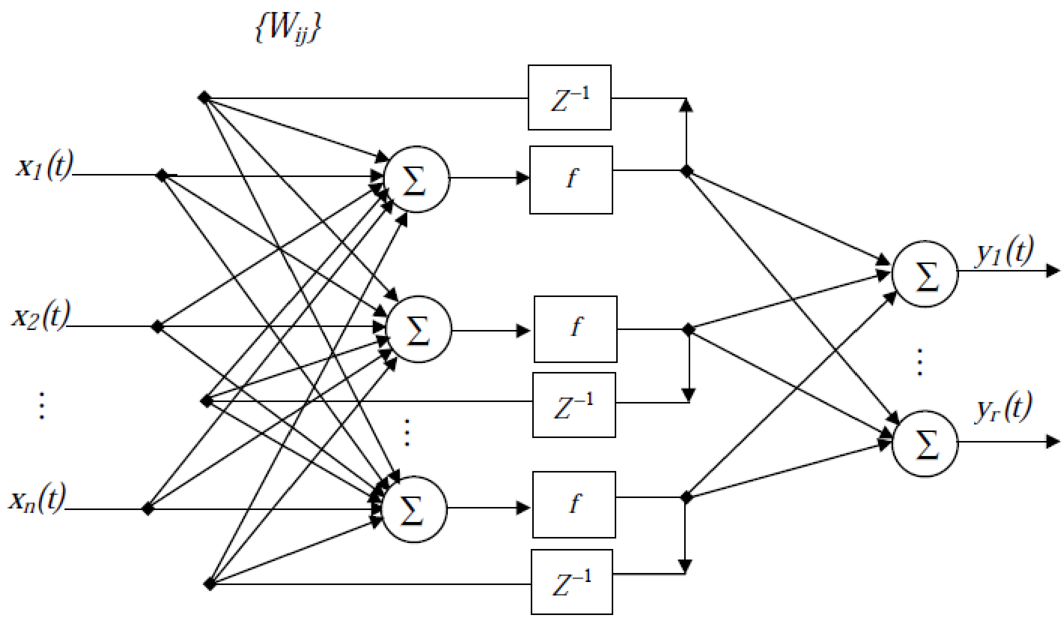
- Estrutura de uma rede neural com recorrência interna local




```
U = X(t,:)';              % entrada externa atual
d = Saida(t,:)';          % saídas desejadas
S = AA*Zold + B*U;  % AA – matriz diagonal
Z = tanh(S);              % entradas para a camada de saída
Y = C*[Z;1];
```

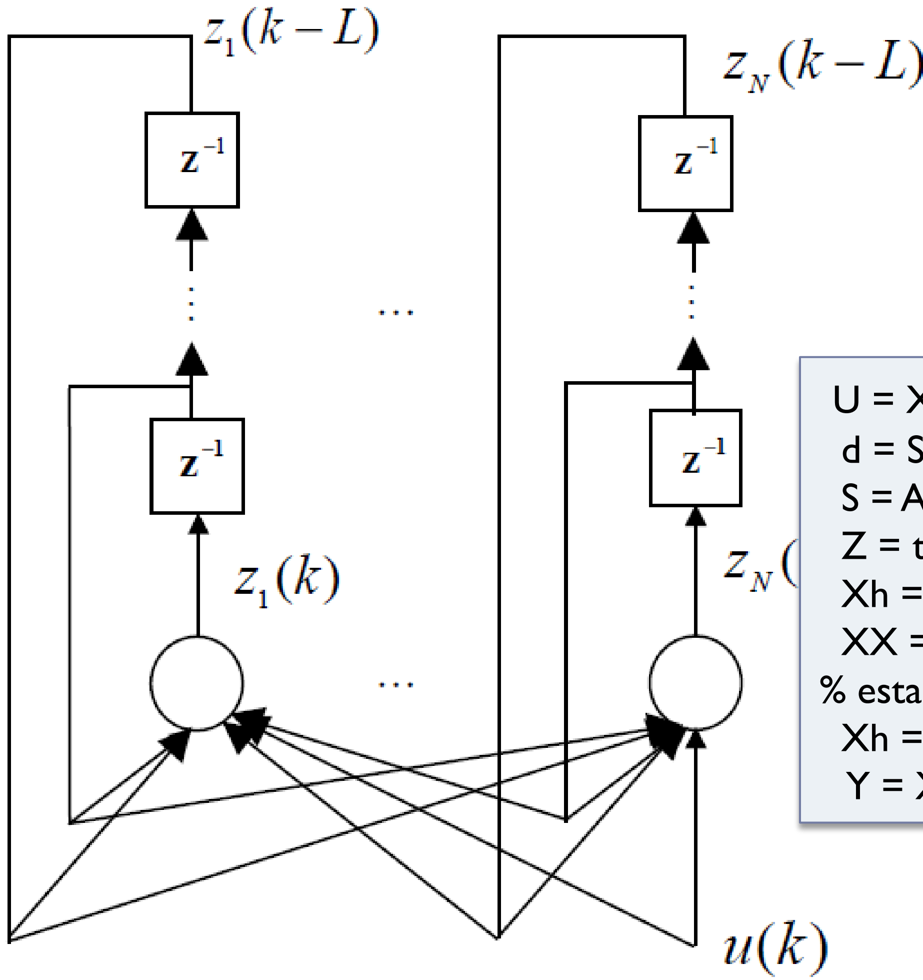
Redes Recorrentes

- Estrutura de uma rede neural com recorrência interna global




```
U = X(t,:)';          % entrada externa atual
 d = Saida(t,:)';     % saídas desejadas
 S = A*Zold + B*U;
 Z = tanh(S);          % entradas para a camada de saída
 Y = C*[Z;1];
```

Excursão ilustrativa por outros tipos de arquiteturas recorrentes

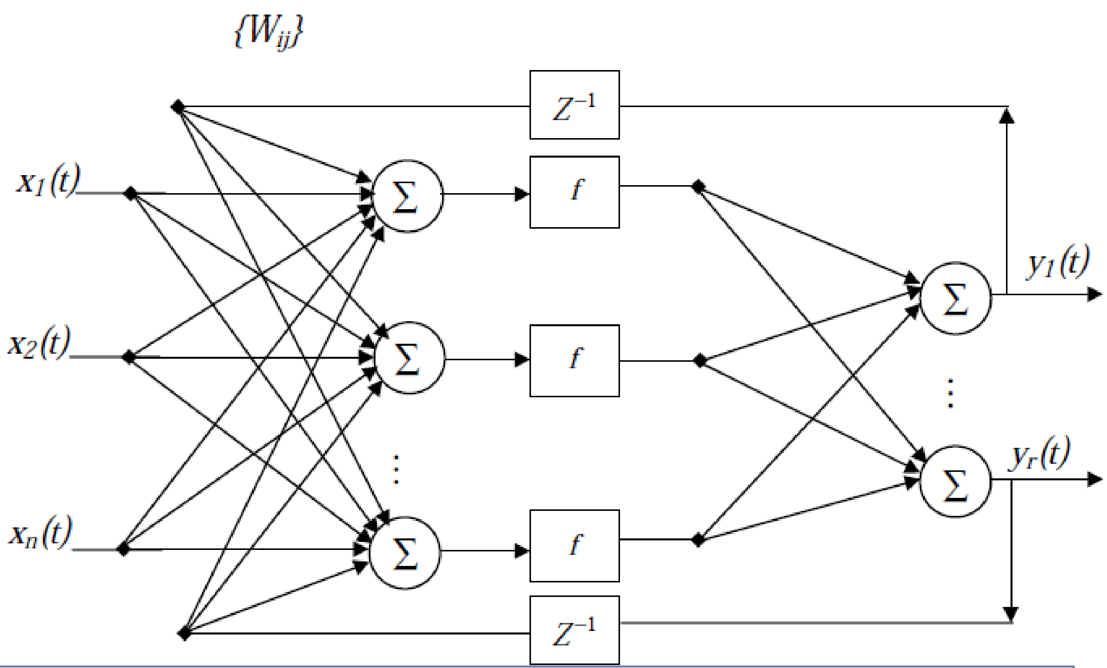



```
U = X(t,:)';        % entrada externa atual
 d = Saida(t,:)';    % saidas desejadas
 S = A*Xold + B*U;
 Z = tanh(S);
 Xh = C*[Z;1];
 XX = [tanh(Xh(1:n(2)); Xh(n(2)+1:end)];
% estados da rede
 Xh = XX(1:n(2));
 Y = XX(n(2)+1:end);                          % saidas
```

- *Redes Neurais totalmente recorrente (FRNN - Fully recurrent neural network)*

Redes Recorrentes

- Estrutura de uma rede neural com recorrência externa




```
U = X(t,:)';      % entrada externa atual
d = Saida(t,:)';      % saidas desejadas
S = A*Yold + B*U;
Z = tanh(S);  % entradas para a camada de saída
Y = C*[Z;1];
```

Paradigmas de identificação de sistemas dinâmicos

- a identificação de sistemas não-lineares pode ser vista como uma concatenação entre um pré-processamento dos dados amostrados, gerando um novo espaço de entrada, e um mapeamento não-linear do novo espaço de entrada para o espaço de saída. A escolha do pré-processamento e a aproximação do mapeamento não-linear de entrada saída podem ser tratadas de forma independente (modelos de entrada-saída) ou conjunta (modelos por espaço de estados).

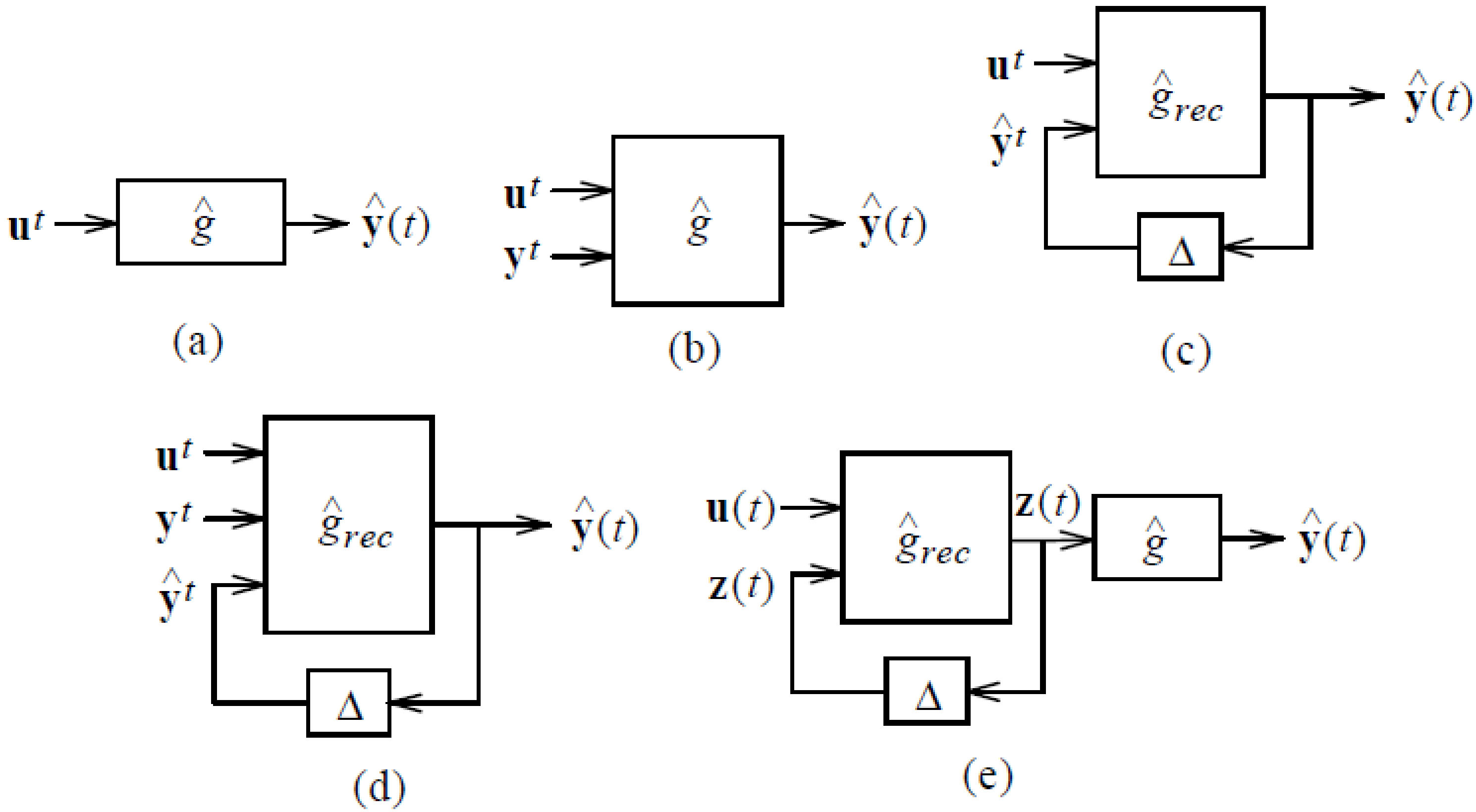
- o novo vetor de entrada $\mathbf{x}(t) \in \Re^m$**,** *resultante do pré-processamento do vetor de* entrada original, é uma função de entradas externas defasadas no tempo, u^{t-1}**,** *de saídas*

- defasadas no tempo, y^{t-1}**,** e *de ruídos aleatórios expressando os resíduos do processo,*de identificação, ε^{t-1}, *na forma:*

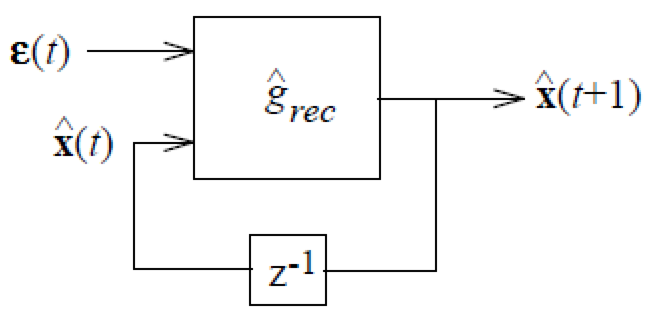
$$\mathbf{x}(t) = \varphi(u^{t-1}, y^{t-1}, \varepsilon^{t-1})$$

Paradigmas de identificação de sistemas dinâmicos

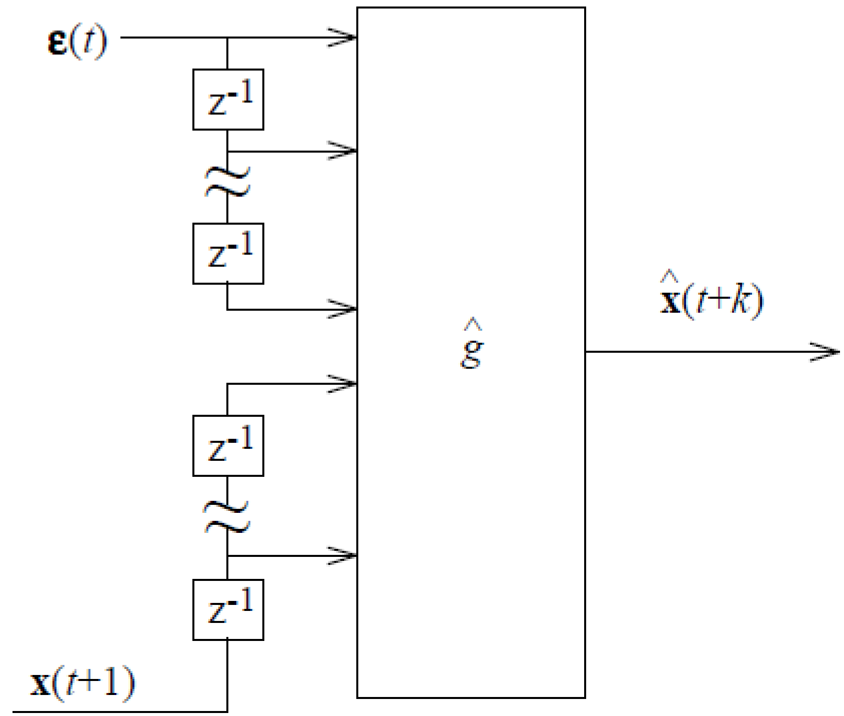
- com isso, o sistema dinâmico da equação (3) pode ser reescrito como segue: $\mathbf{y}(t) = g(\mathbf{x}(t)) + v(t)$



Predição de séries temporais: um passo ou múltiplos passos



(a)



(b)

(a) predição de múltiplos passos à frente

(b) predição de *k passos à frente (k fixo, mas arbitrário)* Uso de linha de derivação de atraso (*tapped-delay line*)

Redes Neurais com Estados de Eco

- Antes de concluirmos o tópico, discutiremos brevemente as Redes Neurais com Estados de Eco (Echo State Networks - ESNs) (JAEGER, 2001), que, em termos simples, procuram unir o "melhor dos dois mundos": aliar o potencial de gerar dinâmica que é próprio de uma rede recorrente a um processo de treinamento cuja complexidade é similar àquela encontrada quando se lida com estruturas *feedforward*.

- Para chegar a um compromisso desse tipo, tais redes empregam duas camadas: uma camada densamente interconectada de elementos de processamento não-linear (EPNL), os quais representam um rico repertório de comportamentos dinâmicos, e uma camada responsável por combinar linearmente as saídas desses EPNLs (que correspondem aos *estados de eco)*.

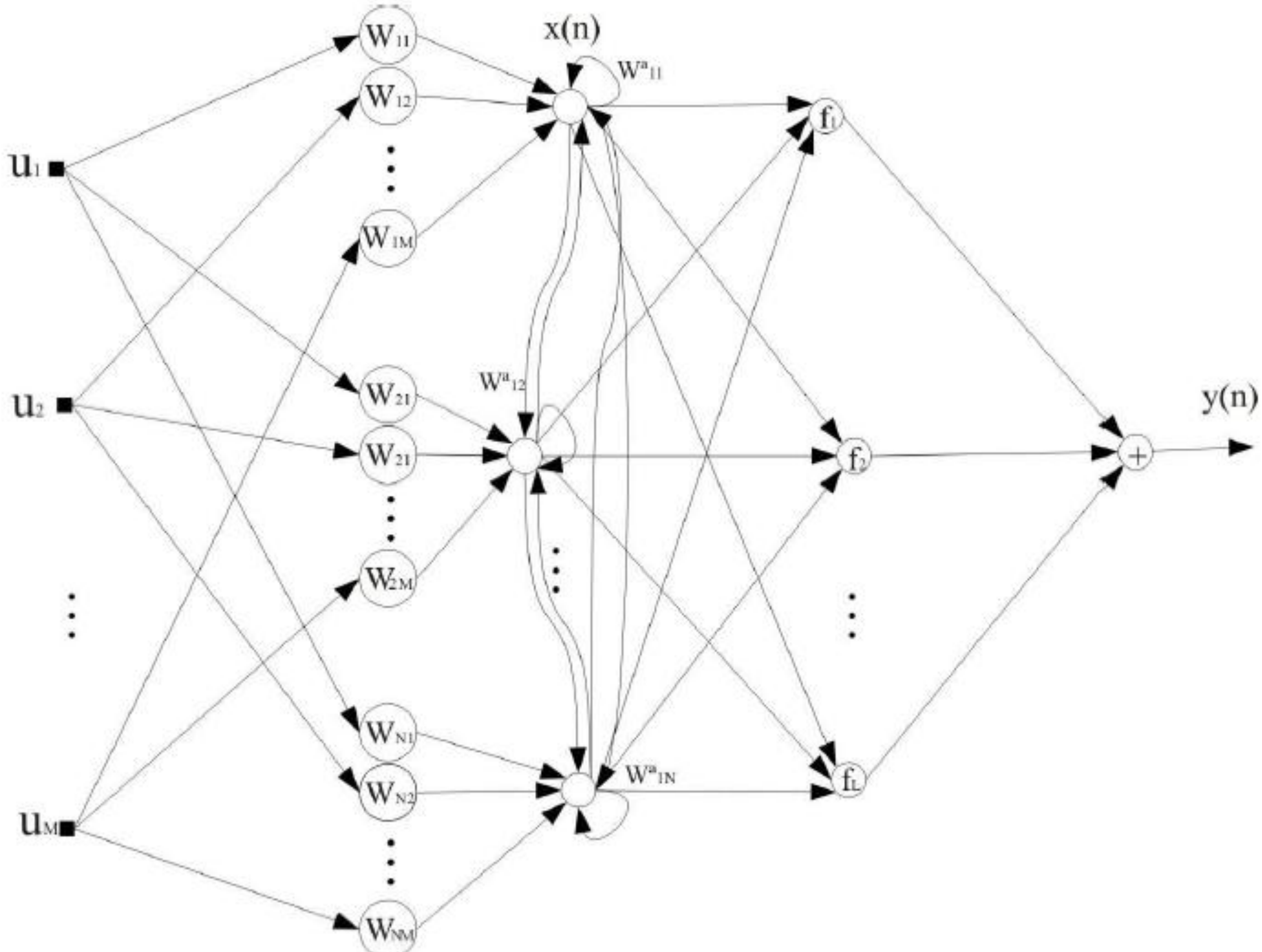
Redes Neurais com Estados de Eco

- O ponto crucial desse paradigma é que apenas os pesos da camada linear são ajustados de acordo com um critério de erro: os pesos da camada não-linear permanecem fixos em valores definidos *a priori*.

- Tal expediente facilita sobremaneira o processo de treinamento, uma vez que o modelo efetivamente adaptado é *linear com respeito aos parâmetros*. *Dessa forma,* é aplicável todo o ferramental por nós visto em tópicos que trataram de filtros de Wiener e de regressão linear.

- Por outro lado, o fato de o "repositório de dinâmicas" da camada não-linear permanecer fixo é algo que pode limitar a capacidade da rede neural no que se refere à possibilidade de aproximar comportamentos dinâmicos genéricos. Isso, aliás, também indica fortemente que a confecção dessa camada é algo que vai requerer uma atenção especial.

Redes Neurais com Estados de Eco



Redes Neurais com Estados de Eco

- O projeto da camada intermediária gravita em torno da noção de gerar um repertório dinâmico suficientemente diversificado e representativo. Essa síntese é auxiliada por diversos conceitos das áreas de processamento de sinais, teoria de informação e aproximação de funções. Na proposta original de Jaeger (JAEGER, 2001), era particularmente enfatizada a estrutura espectral da matriz de pesos dos neurônios da camada não-linear. Num trabalho recente (OZTURK *et al., 2006), mostrou-se que o* emprego de uma medida de entropia (no caso, a entropia de Renyi) pode trazer ganhos significativos em termos do processo de avaliação da "riqueza dinâmica" presente nos estados de eco.