

Memórias Autossociativas de Projeção em Subespaço - SPAM

Emely Pujólli da Silva
Orientador: Prof. Dr. Marcos E. Valle

Universidade Estadual de Campinas
Instituto de Matemática, Matemática Aplicada e Computação Científica

Campinas, 18 de junho de 2015

- 1 Regressão;
- 2 Memórias Associativas;
- 3 Memórias Autoassociativas de Projeção;
- 4 Memórias Autoassociativas de Projeção em Subespaço;
- 5 Relação entre Memórias Autoassociativas de Projeção em Subespaço e Regressão Multilinear;
- 6 Memórias Autoassociativas de Projeção em Subespaço Baseadas em Estimadores Robustos.

Regressão

O objetivo de uma regressão é descrever a relação de fundo (intrínseca) entre uma variável resposta e uma ou mais variáveis independentes.

Uma regressão descreve um conjunto de dados através do ajustamento de funções, sem ter acesso ao processo que geraram os dados.

Regressão

O objetivo de uma regressão é descrever a relação de fundo (intrínseca) entre uma variável resposta e uma ou mais variáveis independentes.

Uma regressão descreve um conjunto de dados através do ajustamento de funções, sem ter acesso ao processo que geraram os dados.

Seja o conjunto de *dados de treino* $\{(x^k, y^k)\}_{k=1}^P$, sendo x^k a variável chamada *independente* e y^k a variável chamada *observação*. Uma **regressão** designa uma aplicação $f : \mathcal{X} \rightarrow \mathcal{Y}$ que descreve a relação entre $x \in \mathcal{X}$ e $y \in \mathcal{Y}$.

Mas não sabemos qual é a relação entre as variáveis, para obtê-la precisamos da ajuda dos estimadores.

Em sua forma mais geral, um estimador f é uma função das observações $f(x^k) = \tilde{y}^k$ que obtém uma estimativa \tilde{y}^k das verdadeiras observações y^k .

Procuramos uma estimativa no sentido que \tilde{y} está próximo de y com grande probabilidade. Há várias formas de fazer isso, uma delas, é avaliar a qualidade da estimativa.

Assumimos que existe uma distribuição de probabilidade $P(x, y)$ em $\mathcal{X} \times \mathcal{Y}$ que padroniza a geração dos dados e a dependência do funcional de fundo [Smola,98]. Tomamos os dados (x^k, y^k) independente e identicamente distribuídos.

Assumimos que existe uma distribuição de probabilidade $P(x, y)$ em $\mathcal{X} \times \mathcal{Y}$ que padroniza a geração dos dados e a dependência do funcional de fundo [Smola,98]. Tomamos os dados (x^k, y^k) independente e identicamente distribuídos.

Nos modelos de predição, o primeiro objetivo é construir um modelo que melhor prediz os valores observados para novos dados.

Assumimos que existe uma distribuição de probabilidade $P(x, y)$ em $\mathcal{X} \times \mathcal{Y}$ que padroniza a geração dos dados e a dependência do funcional de fundo [Smola,98]. Tomamos os dados (x^k, y^k) independente e identicamente distribuídos.

Nos modelos de predição, o primeiro objetivo é construir um modelo que melhor prediz os valores observados para novos dados.

Mas não sabemos a relação entre as associações, então procuramos f de modo a minimizar o risco, ou seja, o erro esperado entre todas as amostras possíveis.

Assumimos que existe uma distribuição de probabilidade $P(x, y)$ em $\mathcal{X} \times \mathcal{Y}$ que padroniza a geração dos dados e a dependência do funcional de fundo [Smola,98]. Tomamos os dados (x^k, y^k) independente e identicamente distribuídos.

Nos modelos de predição, o primeiro objetivo é construir um modelo que melhor prediz os valores observados para novos dados.

Mas não sabemos a relação entre as associações, então procuramos f de modo a minimizar o risco, ou seja, o erro esperado entre todas as amostras possíveis. Em particular utilizamos a forma de risco empírico para encontrar a estimativa f .

Definição (Risco Empírico)

Denote por $(x, y, f(x)) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ a tripla consistindo de um padrão x , uma observação y e uma previsão $f(x)$. O **risco empírico** é definido como

$$R_{emp}[f] := \frac{1}{n} \sum_{i=1}^n \rho(x_i, y_i, f(x_i)) \quad (1)$$

onde a função $\rho : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \rightarrow [0, \infty)$ possui a propriedade $\rho(x, y, y) = 0$ para todo $x \in \mathcal{X}$ e $y \in \mathcal{Y}$ será chamada de **função perda** [Smola,98].

Definição (Risco Empírico)

Denote por $(x, y, f(x)) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ a tripla consistindo de um padrão x , uma observação y e uma previsão $f(x)$. O **risco empírico** é definido como

$$R_{emp}[f] := \frac{1}{n} \sum_{i=1}^n \rho(x_i, y_i, f(x_i)) \quad (1)$$

onde a função $\rho : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \rightarrow [0, \infty)$ possui a propriedade $\rho(x, y, y) = 0$ para todo $x \in \mathcal{X}$ e $y \in \mathcal{Y}$ será chamada de **função perda** [Smola,98].

Uma estimativa é construída sobre a abordagem de perda e risco, nesse sentido, a função perda representa o preço que dispomos a pagar na previsão $f(x)$ em vez de y . Ou seja, um estimador pode ou não ser penalizado ao tomar uma estimativa muito próxima da observação.

Observe que a expressão do risco empírico pode ser calculada para determinadas funções perda. Resolver esse problema constitui uma classe de estimadores conhecidos como **M-Estimador** na literatura, expressos pela relação

$$\min R_{emp}[f] = \min \frac{1}{n} \sum_{i=1}^n \rho(x_i, y_i, f(x_i)) \quad (2)$$

Observe que a expressão do risco empírico pode ser calculada para determinadas funções perda. Resolver esse problema constitui uma classe de estimadores conhecidos como **M-Estimador** na literatura, expressos pela relação

$$\min R_{emp}[f] = \min \frac{1}{n} \sum_{i=1}^n \rho(x_i, y_i, f(x_i)) \quad (2)$$

Existem muitas funções que descrevem bem os dados, e encontrá-las nem sempre é fácil. Por isso restringimos ao caso em que a relação entre os dados é assumida linear.

O Caso da Regressão Multilinear

Procuramos, em particular, solucionar um problema de regressão multilinear.

O Caso da Regressão Multilinear

Procuramos, em particular, solucionar um problema de regressão multilinear.

No problema de regressão multilinear pretende-se modelar uma variável resposta $\mathbf{y} = [y_1, \dots, y_n]$ com base em p variáveis independentes $\mathbf{x}^1, \dots, \mathbf{x}^p$. Temos n conjuntos de observações, isto é, $(x_i^1, x_i^2, \dots, x_i^p, y_i)$ com $i = 1, \dots, n$. Admitindo que a relação entre a variável independente e a observação é linear, temos

$$f(x_i) = \alpha_0 + \alpha_1 x_i^1 + \alpha_2 x_i^2 + \dots + \alpha_p x_i^p + \varepsilon_i, \quad i = 1, \dots, n$$

O Caso da Regressão Multilinear

Na forma matricial

$$\mathbf{y} = X\boldsymbol{\alpha} + \boldsymbol{\varepsilon}, \quad (3)$$

com

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_1^1 & \dots & x_1^p \\ 1 & x_2^1 & \dots & x_2^p \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n^1 & \dots & x_n^p \end{bmatrix}, \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix} \quad \text{e} \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

representando o vetor das observações, a matriz das amostras, o vetor dos coeficientes e o vetor dos erros, respectivamente.

O Caso da Regressão Multilinear

Na forma matricial

$$\mathbf{y} = X\boldsymbol{\alpha} + \boldsymbol{\varepsilon}, \quad (3)$$

com

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_1^1 & \dots & x_1^p \\ 1 & x_2^1 & \dots & x_2^p \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n^1 & \dots & x_n^p \end{bmatrix}, \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix} \quad \text{e} \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

representando o vetor das observações, a matriz das amostras, o vetor dos coeficientes e o vetor dos erros, respectivamente.

Neste caso, os estimadores procuram o vetor $\boldsymbol{\alpha}$ de modo a aproximar $X\boldsymbol{\alpha}$ de \mathbf{y} , isto é $X\boldsymbol{\alpha} \approx \mathbf{y}$, ou ainda $f(X) \approx \mathbf{y}$.

O Caso da Regressão Multilinear

No caso de uma regressão, f é uma função que estima quantidades reais. Dessa forma, o que importa é o tamanho da diferença $y - f(x)$, ou seja, a quantidade de previsão errada.

O Caso da Regressão Multilinear

No caso de uma regressão, f é uma função que estima quantidades reais. Dessa forma, o que importa é o tamanho da diferença $y - f(x)$, ou seja, a quantidade de previsão errada.

A função perda definida anteriormente para o caso da regressão multilinear toma a forma

$$\rho(\mathbf{x}, \mathbf{y}, f(\mathbf{x})) = \ell(\mathbf{y} - X\alpha) \quad (4)$$

com $\ell : \mathbb{R} \rightarrow [0, \infty)$.

O Caso da Regressão Multilinear

No caso de uma regressão, f é uma função que estima quantidades reais. Dessa forma, o que importa é o tamanho da diferença $y - f(x)$, ou seja, a quantidade de previsão errada.

A função perda definida anteriormente para o caso da regressão multilinear toma a forma

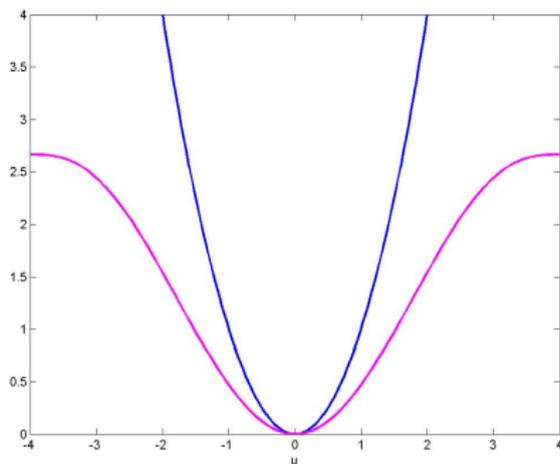
$$\rho(\mathbf{x}, \mathbf{y}, f(\mathbf{x})) = \ell(\mathbf{y} - X\boldsymbol{\alpha}) \quad (4)$$

com $\ell : \mathbb{R} \rightarrow [0, \infty)$.

Portanto um estimador de regressão multilinear encontra o melhor coeficiente $\boldsymbol{\alpha}^*$ que minimiza o risco empírico, ou seja

$$\boldsymbol{\alpha}^* = \arg \min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n \ell \left(y_i - \sum_{k=1}^p x_i^k \alpha_k \right). \quad (5)$$

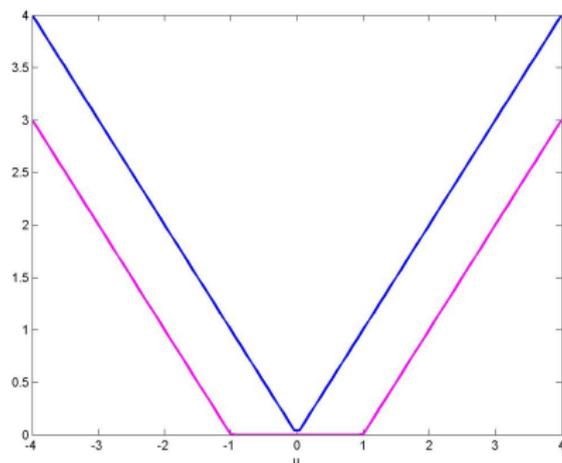
Exemplos de Função Perda



Função perda quadrática: $\ell_Q(u) = u^2$,

Função de Tukey: $\ell_T(u) = \begin{cases} \frac{u^2}{2} - \frac{u^4}{2k^2} + \frac{u^6}{6k^4} & \text{se } |u| \leq k \\ \frac{k^2}{6} & \text{se } |u| > k \end{cases}$

Exemplos de Função Perda



Função módulo: $l_1(u) = |u|$,

Função perda ϵ -insensível: $l_\epsilon(u) = \begin{cases} |u| - \epsilon & \text{se } |u| \geq \epsilon \\ 0 & \text{c. c.} \end{cases}$

Estimadores Robustos

Como a expressão do risco empírico depende da função perda, e essa é escolhida de acordo com o problema a ser resolvido, a solução da minimização do risco pode ser feita utilizando diversos algoritmos, dentre esses estão os métodos iterativos e de otimização convexa.

Estimadores Robustos

Como a expressão do risco empírico depende da função perda, e essa é escolhida de acordo com o problema a ser resolvido, a solução da minimização do risco pode ser feita utilizando diversos algoritmos, dentre esses estão os métodos iterativos e de otimização convexa.

Utilizamos duas medidas para escolher os estimadores:

Estimadores Robustos

Como a expressão do risco empírico depende da função perda, e essa é escolhida de acordo com o problema a ser resolvido, a solução da minimização do risco pode ser feita utilizando diversos algoritmos, dentre esses estão os métodos iterativos e de otimização convexa.

Utilizamos duas medidas para escolher os estimadores:

- * Robustez é a insensibilidade à pequenas variações nas hipóteses.
- * O *ponto de quebra* ou *breakdown point* é a menor fração de contaminação que pode causar o estimador $f(x^k)$ tomar valores arbitrariamente longe de y^k [Rousseeuw,87].

Estimadores Robustos

Como a expressão do risco empírico depende da função perda, e essa é escolhida de acordo com o problema a ser resolvido, a solução da minimização do risco pode ser feita utilizando diversos algoritmos, dentre esses estão os métodos iterativos e de otimização convexa.

Utilizamos duas medidas para escolher os estimadores:

- * Robustez é a insensibilidade à pequenas variações nas hipóteses.
- * O *ponto de quebra* ou *breakdown point* é a menor fração de contaminação que pode causar o estimador $f(x^k)$ tomar valores arbitrariamente longe de y^k [Rousseeuw,87].

Entretanto, tanto os dados $\mathbf{x}^1, \dots, \mathbf{x}^p$ quanto \mathbf{y} podem ser corrompidos, quando isso ocorre denominamos por *leverage points* e *outliers*, respectivamente.

Memórias Associativas

Baseados na forma em que o cérebro é capaz de armazenar e recordar informações, apresentamos as **Memórias Associativas**.

Memórias Associativas

Baseados na forma em que o cérebro é capaz de armazenar e recordar informações, apresentamos as **Memórias Associativas**.

Uma memória associativa é um sistema de entrada e saída capaz de armazenar e recordar um conjunto finito de dados. É esperado que a memória seja capaz de recuperar a informação como resposta mesmo quando apresentado na entrada com um dado corrompido. São duas as fases envolvidas na operação de uma memória associativa:

Memórias Associativas

Baseados na forma em que o cérebro é capaz de armazenar e recordar informações, apresentamos as **Memórias Associativas**.

Uma memória associativa é um sistema de entrada e saída capaz de armazenar e recordar um conjunto finito de dados. É esperado que a memória seja capaz de recuperar a informação como resposta mesmo quando apresentado na entrada com um dado corrompido. São duas as fases envolvidas na operação de uma memória associativa:

- *Fase de armazenamento*: o processo de sintetizar uma memória associativa;

Memórias Associativas

Baseados na forma em que o cérebro é capaz de armazenar e recordar informações, apresentamos as **Memórias Associativas**.

Uma memória associativa é um sistema de entrada e saída capaz de armazenar e recordar um conjunto finito de dados. É esperado que a memória seja capaz de recuperar a informação como resposta mesmo quando apresentado na entrada com um dado corrompido. São duas as fases envolvidas na operação de uma memória associativa:

- *Fase de armazenamento*: o processo de sintetizar uma memória associativa;
- *Fase de recordação*: a memória é testada para verificar se foram armazenadas corretamente e a capacidade de correção de erro é medida.

Memórias Associativas

Formalmente seja $\mathcal{F} = \{(\mathbf{u}^k, \mathbf{v}^k) : k = 1, \dots, p\}$, composto dos vetores $\mathbf{u}^k \in \mathbb{R}^n$ e $\mathbf{v}^k \in \mathbb{R}^m$, chamado *conjunto das memórias fundamentais*. Para uma memória associativa criamos uma associação $\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ entre a entrada \mathbf{u}^k e a saída \mathbf{v}^k , chamada aplicação associativa, com a propriedade de que

$$\mathcal{M}(\mathbf{u}^k) = \mathbf{v}^k \text{ e } \mathcal{M}(\tilde{\mathbf{u}}^k) = \mathbf{v}^k \quad \forall k = 1, \dots, p \quad (6)$$

ou seja, $\tilde{\mathbf{u}}^k$ e \mathbf{u}^k devem produzir a mesma saída.

Cada par $(\mathbf{u}^k, \mathbf{v}^k)$ armazenado na memória é chamado *associação* com a entrada do sistema \mathbf{u}^k dito *padrão-chave* e a saída do sistema \mathbf{v}^k dito *padrão recordado*.

Memórias Associativas

As memórias associativas que são classificadas de várias formas, em particular, dizemos que uma memória associativa é:

Memórias Associativas

As memórias associativas que são classificadas de várias formas, em particular, dizemos que uma memória associativa é:

- *Heteroassociativa* quando a entrada \mathbf{u}^k é diferente da saída \mathbf{v}^k ;

Memórias Associativas

As memórias associativas que são classificadas de várias formas, em particular, dizemos que uma memória associativa é:

- *Heteroassociativa* quando a entrada \mathbf{u}^k é diferente da saída \mathbf{v}^k ;
- *Autoassociativa* no caso em que a entrada e a saída são os mesmos. Neste caso, os termos memórias fundamentais e chave fundamental são sinônimos.

Memórias Associativas

As memórias associativas que são classificadas de várias formas, em particular, dizemos que uma memória associativa é:

- *Heteroassociativa* quando a entrada \mathbf{u}^k é diferente da saída \mathbf{v}^k ;
- *Autoassociativa* no caso em que a entrada e a saída são os mesmos. Neste caso, os termos memórias fundamentais e chave fundamental são sinônimos.

Pelas variedades de aplicações e por ser conceitualmente mais simples, faremos o estudo das memórias autoassociativas.

Memórias Autoassociativas

Se a memória é autoassociativa o conjunto das memórias fundamentais é da forma

$$\mathcal{F} = \{\mathbf{u}^1, \mathbf{u}^2, \dots, \mathbf{u}^p\},$$

com os vetores $\mathbf{u}^1, \dots, \mathbf{u}^p \in \mathbb{R}^n$ denominados *memórias fundamentais*.

Definimos uma memória autoassociativa como uma aplicação

$\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ tal que

$$\mathcal{M}(\mathbf{u}^k) = \mathbf{u}^k \quad \text{e} \quad \mathcal{M}(\tilde{\mathbf{u}}^k) = \mathbf{u}^k \quad \forall k = 1, \dots, p \quad (7)$$

para $\tilde{\mathbf{u}}^k$ versão ruidosa de \mathbf{u}^k . A aplicação \mathcal{M} é chamada *aplicação associativa*.

Memórias Autoassociativas

Se a memória é autoassociativa o conjunto das memórias fundamentais é da forma

$$\mathcal{F} = \{\mathbf{u}^1, \mathbf{u}^2, \dots, \mathbf{u}^p\},$$

com os vetores $\mathbf{u}^1, \dots, \mathbf{u}^p \in \mathbb{R}^n$ denominados *memórias fundamentais*.

Definimos uma memória autoassociativa como uma aplicação

$\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ tal que

$$\mathcal{M}(\mathbf{u}^k) = \mathbf{u}^k \quad \text{e} \quad \mathcal{M}(\tilde{\mathbf{u}}^k) = \mathbf{u}^k \quad \forall k = 1, \dots, p \quad (7)$$

para $\tilde{\mathbf{u}}^k$ versão ruidosa de \mathbf{u}^k . A aplicação \mathcal{M} é chamada *aplicação associativa*.

Assumimos as memórias fundamentais linearmente independentes.

Memória Associativa Linear - LAM

Os primeiros modelos de memórias por associação, assume-se que os padrões são transformados linearmente [Kohonen,87] através da equação

$$\mathbf{u}^k = M\mathbf{u}^k \quad \forall k = 1, \dots, p, \quad (8)$$

onde $M \in \mathbb{R}^{m \times n}$ representa a matriz de interconexões. Essa abordagem é conhecida por *Memória Associativa Linear* (LAM) e classicamente, este modelo é referenciado à Anderson, Kohonen e Nakano.

Memória Autoassociativa Linear Ótima - OLAM

Em particular, procura-se a matriz M tal que $U = MU$, com $U = [\mathbf{u}^1, \dots, \mathbf{u}^p]$ chamada matriz das memórias fundamentais.

Memória Autoassociativa Linear Ótima - OLAM

Em particular, procura-se a matriz M tal que $U = MU$, com $U = [\mathbf{u}^1, \dots, \mathbf{u}^p]$ chamada matriz das memórias fundamentais.

A expressão possui solução do problema de quadrados mínimos com a norma de Frobenius para matrizes [Watkins,02],[Hecht-Nielsen,89], que é expressa da seguinte forma

$$\min_{M \in \mathbb{R}^{m \times n}} \|U - MU\|_F^2. \quad (9)$$

Logo, podemos resolver (9) e encontrar a matriz de norma mínima M_O^* que resolve o problema da memória (8).

Memória Autoassociativa Linear Ótima - OLAM

A solução

$$M_O^* = UU^\dagger \quad (10)$$

em que U^\dagger é conhecida como a pseudo-inversa (ou inversa generalizada de Moore-Penrose) de U e pode ser calculada utilizando

$$U^\dagger = (U^T U)^{-1} U^T.$$

Dado a entrada $\mathbf{x} \in \mathbb{R}^n$, a memória

$$\mathcal{M}_O(\mathbf{x}) = M_O^* \mathbf{x},$$

constitui o modelo de **Memória Autoassociativa Linear Ótima (OLAM)** [Kohonen,87],[Hassoun,95].

Memória Autoassociativa Linear Ótima - OLAM

A solução

$$M_O^* = UU^\dagger \quad (10)$$

em que U^\dagger é conhecida como a pseudo-inversa (ou inversa generalizada de Moore-Penrose) de U e pode ser calculada utilizando

$$U^\dagger = (U^T U)^{-1} U^T.$$

Dado a entrada $\mathbf{x} \in \mathbb{R}^n$, a memória

$$\mathcal{M}_O(\mathbf{x}) = M_O^* \mathbf{x},$$

constitui o modelo de **Memória Autoassociativa Linear Ótima (OLAM)** [Kohonen,87],[Hassoun,95].

A matriz M é a matriz de projeção no espaço gerado pelos padrões fundamentais \mathbf{u}^k , por isso dizemos que esta é uma memória armazenada por projeção.

Exemplo - OLAM

Seja $\mathcal{F} = \{\mathbf{u}^1, \mathbf{u}^2, \mathbf{u}^3, \mathbf{u}^4\} \subseteq \mathbb{R}^6$ o conjunto das memórias fundamentais composto dos vetores

$$\mathbf{u}^1 = \begin{bmatrix} 10 \\ 2 \\ 9 \\ 4 \\ 10 \\ 4 \end{bmatrix}, \mathbf{u}^2 = \begin{bmatrix} 3 \\ 4 \\ 4 \\ 2 \\ 5 \\ 3 \end{bmatrix}, \mathbf{u}^3 = \begin{bmatrix} 6 \\ 9 \\ 4 \\ 9 \\ 4 \\ 8 \end{bmatrix}, \text{ e } \mathbf{u}^4 = \begin{bmatrix} 8 \\ 6 \\ 3 \\ 4 \\ 3 \\ 10 \end{bmatrix}. \quad (11)$$

Tomamos o vetor de entrada

$$\mathbf{x} = [0 \ 2 \ 9 \ 4 \ 10 \ 8]^T, \quad (12)$$

obtido através da soma do vetor de ruído $[-10 \ 0 \ 0 \ 0 \ 0 \ 4]^T$ e \mathbf{u}^1 .

Exemplo - OLAM

Calculamos a matriz da projeção

$$M_0 = U * U^\dagger$$
$$= \begin{bmatrix} 0.69 & -0.28 & 0.15 & 0.12 & 0.05 & 0.30 \\ -0.28 & 0.72 & 0.02 & 0.13 & 0.14 & 0.28 \\ 0.15 & 0.02 & 0.35 & 0.04 & 0.43 & -0.09 \\ 0.12 & 0.13 & 0.04 & 0.92 & -0.11 & -0.13 \\ 0.05 & 0.14 & 0.43 & -0.11 & 0.60 & -0.09 \\ 0.30 & 0.28 & -0.09 & -0.13 & -0.09 & 0.69 \end{bmatrix}$$

e recuperamos a memória

$$\mathcal{M}_O(\mathbf{x}) = [4.30 \quad 5.96 \quad 7.07 \quad 2.22 \quad 9.05 \quad 3.76]^T. \quad (13)$$

A saída $\mathcal{M}_O(\mathbf{x})$ não recupera a memória fundamental \mathbf{u}^1 corretamente.

Exemplo - OLAM

Calculamos a matriz da projeção

$$M_0 = U * U^\dagger$$
$$= \begin{bmatrix} 0.69 & -0.28 & 0.15 & 0.12 & 0.05 & 0.30 \\ -0.28 & 0.72 & 0.02 & 0.13 & 0.14 & 0.28 \\ 0.15 & 0.02 & 0.35 & 0.04 & 0.43 & -0.09 \\ 0.12 & 0.13 & 0.04 & 0.92 & -0.11 & -0.13 \\ 0.05 & 0.14 & 0.43 & -0.11 & 0.60 & -0.09 \\ 0.30 & 0.28 & -0.09 & -0.13 & -0.09 & 0.69 \end{bmatrix}$$

e recuperamos a memória

$$\mathcal{M}_O(\mathbf{x}) = [4.30 \ 5.96 \ 7.07 \ 2.22 \ 9.05 \ 3.76]^T. \quad (13)$$

A saída $\mathcal{M}_O(\mathbf{x})$ não recupera a memória fundamental \mathbf{u}^1 corretamente. Ainda, o erro $\|\mathbf{u}^1 - \mathcal{M}_O(\mathbf{x})\|_2 = 7.48$ é menor que o erro $\|\mathbf{u}^1 - \mathbf{x}\|_2 = 10.77$.

Memória Autoassociativa de Projeção - PAM

Motivados pelas propriedades e características atrativas buscamos definir as memórias de projeção para o caso autoassociativo.

Memória Autoassociativa de Projeção - PAM

Motivados pelas propriedades e características atrativas buscamos definir as memórias de projeção para o caso autoassociativo.

Observe que a memória fundamental \mathbf{u}^k é um ponto fixo da memória autoassociativa.

Memória Autoassociativa de Projeção - PAM

Motivados pelas propriedades e características atrativas buscamos definir as memórias de projeção para o caso autoassociativo.

Observe que a memória fundamental \mathbf{u}^k é um ponto fixo da memória autoassociativa.

Considere

$$\mathcal{S} = \{\mathbf{s} \in \mathbb{R}^n : \mathcal{M}(\mathbf{s}) = \mathbf{s}\},$$

o conjunto dos pontos fixos da aplicação \mathcal{M} . Note que se \mathcal{M} é uma memória autoassociativa então $\mathcal{F} \subseteq \mathcal{S}$.

Memória Autoassociativa de Projeção - PAM

Motivados pelas propriedades e características atrativas buscamos definir as memórias de projeção para o caso autoassociativo.

Observe que a memória fundamental \mathbf{u}^k é um ponto fixo da memória autoassociativa.

Considere

$$\mathcal{S} = \{\mathbf{s} \in \mathbb{R}^n : \mathcal{M}(\mathbf{s}) = \mathbf{s}\},$$

o conjunto dos pontos fixos da aplicação \mathcal{M} . Note que se \mathcal{M} é uma memória autoassociativa então $\mathcal{F} \subseteq \mathcal{S}$.

É natural procurar a saída $\mathcal{M}(\mathbf{u}^k)$ como a projeção do vetor de entrada no conjunto \mathcal{S} para alguma distância.

Memória Autoassociativa de Projeção - PAM

Dessa forma, combinando a definição de memória como função associativa e o conjunto dos pontos fixos da aplicação, o resultado é a seguinte classe de memórias.

Memória Autoassociativa de Projeção - PAM

Dessa forma, combinando a definição de memória como função associativa e o conjunto dos pontos fixos da aplicação, o resultado é a seguinte classe de memórias.

Definição (Memória Autoassociativa de Projeção)

Uma memória associativa $\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ é uma Memória Autoassociativa de Projeção (PAM), se para qualquer entrada $\mathbf{x} \in \mathbb{R}^n$ temos que

$$\mathcal{M}(\mathbf{x}) \in \mathcal{S} \text{ e } d(\mathbf{x}, \mathcal{M}(\mathbf{x})) \leq d(\mathbf{x}, \mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S} \quad (14)$$

com d uma certa função de distância em \mathbb{R}^n .

Memória Autoassociativa de Projeção - PAM

A PAM é uma interessante classe de memórias pois possui as seguintes propriedades:

Memória Autoassociativa de Projeção - PAM

A PAM é uma interessante classe de memórias pois possui as seguintes propriedades:

- 1 O vetor $\mathcal{M}(\mathbf{x})$ recuperado pelo modelo PAM é a melhor aproximação de \mathbf{x} em \mathcal{S} .

Memória Autoassociativa de Projeção - PAM

A PAM é uma interessante classe de memórias pois possui as seguintes propriedades:

- 1 O vetor $\mathcal{M}(\mathbf{x})$ recuperado pelo modelo PAM é a melhor aproximação de \mathbf{x} em \mathcal{S} .
- 2 A saída se mantém inalterada sobre repetidas aplicações de \mathcal{M} .

Memória Autoassociativa de Projeção - PAM

A PAM é uma interessante classe de memórias pois possui as seguintes propriedades:

- 1 O vetor $\mathcal{M}(\mathbf{x})$ recuperado pelo modelo PAM é a melhor aproximação de \mathbf{x} em \mathcal{S} .
- 2 A saída se mantém inalterada sobre repetidas aplicações de \mathcal{M} .
- 3 A PAM exhibe capacidade ótima de armazenamento (assumindo $\mathcal{F} \subseteq \mathcal{S}$).

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Motivados pelo fato que muitos grupos de transformações consideram a saída como combinação linear dos vetores da base.

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Motivados pelo fato que muitos grupos de transformações consideram a saída como combinação linear dos vetores da base.

Colocamos \mathcal{S} como subespaço e assim impomos que a memória seja invariante a escala. O resultado é a próxima classe de memórias.

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Motivados pelo fato que muitos grupos de transformações consideram a saída como combinação linear dos vetores da base.

Colocamos \mathcal{S} como subespaço e assim impomos que a memória seja invariante a escala. O resultado é a próxima classe de memórias.

Definição (Memória Autoassociativa de Projeção em Subespaço)

Se $\mathcal{M} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ é uma memória autoassociativa de projeção e \mathcal{S} é subespaço (linear) de \mathbb{R}^n , então dizemos que \mathcal{M} é uma Memória Autoassociativa de Projeção em Subespaço (SPAM).

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Como menor conjunto em \mathcal{S} contendo todas as memórias fundamentais é \mathcal{F} , colocamos a seguinte hipótese:

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Como menor conjunto em \mathcal{S} contendo todas as memórias fundamentais é \mathcal{F} , colocamos a seguinte hipótese:

“O conjunto dos pontos fixos da aplicação associativa \mathcal{M} é subespaço gerado pelas memórias fundamentais.”

Memórias Autoassociativas de Projeção em Subespaço - SPAM

Como menor conjunto em \mathcal{S} contendo todas as memórias fundamentais é \mathcal{F} , colocamos a seguinte hipótese:

“O conjunto dos pontos fixos da aplicação associativa \mathcal{M} é subespaço gerado pelas memórias fundamentais.”

Para reduzir a quantidade de memórias espúrias, que são as associações que não fazem parte do conjunto das memórias fundamentais.

Dado a entrada $\mathbf{x} \in \mathbb{R}^n$ caracterizamos a saída $\mathcal{M}(\mathbf{x}) \in \mathcal{S}$ como a combinação linear das memórias fundamentais $\mathbf{u}^1, \dots, \mathbf{u}^p$. Em outros termos, existem $\alpha_1, \dots, \alpha_p \in \mathbb{R}$ escalares tais que

$$\mathcal{M}(\mathbf{x}) = \alpha_1 \mathbf{u}^1 + \alpha_2 \mathbf{u}^2 + \dots + \alpha_p \mathbf{u}^p. \quad (15)$$

Havendo os vetores das memórias fundamentais \mathcal{F} , para descrever a aplicação associativa \mathcal{M} precisamos encontrar os coeficientes $\alpha_1, \dots, \alpha_p$.

Dado a entrada $\mathbf{x} \in \mathbb{R}^n$ caracterizamos a saída $\mathcal{M}(\mathbf{x}) \in \mathcal{S}$ como a combinação linear das memórias fundamentais $\mathbf{u}^1, \dots, \mathbf{u}^p$. Em outros termos, existem $\alpha_1, \dots, \alpha_p \in \mathbb{R}$ escalares tais que

$$\mathcal{M}(\mathbf{x}) = \alpha_1 \mathbf{u}^1 + \alpha_2 \mathbf{u}^2 + \dots + \alpha_p \mathbf{u}^p. \quad (15)$$

Havendo os vetores das memórias fundamentais \mathcal{F} , para descrever a aplicação associativa \mathcal{M} precisamos encontrar os coeficientes $\alpha_1, \dots, \alpha_p$.

Matricialmente a expressão (15) toma a forma

$$\mathcal{M}(\mathbf{x}) = U\alpha, \quad (16)$$

sendo $U = [\mathbf{u}^1, \mathbf{u}^2, \dots, \mathbf{u}^p] \in \mathbb{R}^{n \times p}$ a chamada *matriz das memórias fundamentais* e $\alpha = [\alpha_1, \dots, \alpha_p]^T$ o vetor dos coeficientes.

Relação Entre SPAM e Regressão Multilinear

Considere a segunda restrição da definição de PAM e a equação da memória

$$d(\mathbf{x}, \mathcal{M}(\mathbf{x})) \leq d(\mathbf{x}, \mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$$

Relação Entre SPAM e Regressão Multilinear

Considere a segunda restrição da definição de PAM e a equação da memória

$$d(\mathbf{x}, U\alpha) \leq d(\mathbf{x}, \mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$$

Relação Entre SPAM e Regressão Multilinear

Considere a segunda restrição da definição de PAM e a equação da memória

$$d(\mathbf{x}, U\alpha) \leq d(\mathbf{x}, \mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$$

procuramos encontrar $\alpha^*(\mathbf{x})$ que minimize a expressão.

$$\alpha^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} d(U\alpha, \mathbf{x}).$$

Relação Entre SPAM e Regressão Multilinear

Considere a segunda restrição da definição de PAM e a equação da memória

$$d(\mathbf{x}, U\alpha) \leq d(\mathbf{x}, \mathbf{s}), \quad \forall \mathbf{s} \in \mathcal{S}$$

procuramos encontrar $\alpha^*(\mathbf{x})$ que minimize a expressão.

$$\alpha^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} d(U\alpha, \mathbf{x}).$$

Este também pode ser visto como o problema de encontrar α de forma a aproximar $U\alpha$ de \mathbf{x} , isto é,

$$\mathbf{x} \approx U\alpha. \quad (17)$$

que pode ser visto como um problema de regressão multilinear já que o conjunto das memórias fundamentais \mathcal{F} possui uma relação intrínseca, ou seja são dados de treino.

Relação Entre SPAM e Regressão Multilinear

Considerando a distância

$$\alpha^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} d(U\alpha, \mathbf{x}). \quad (18)$$

em termos da função perda, ou seja

$$d(U\alpha, \mathbf{x}) = \sum_{i=1}^n \ell \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right). \quad (19)$$

Obtemos o problema de estimar os coeficientes da regressão multilinear

$$\alpha^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} \sum_{i=1}^n \ell \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right),$$

e podemos utilizar a classe de M-estimadores para encontrar a melhor estimativa para o coeficiente α .

Obtemos o problema de estimar os coeficientes da regressão multilinear

$$\alpha^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} \sum_{i=1}^n \ell \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right),$$

e podemos utilizar a classe de M-estimadores para encontrar a melhor estimativa para o coeficiente α .

Essa é a abordagem proposta: construir memórias autoassociativas de projeção em subespaço para diferentes funções distância.

A OLAM é uma SPAM

Vamos retomar o caso da OLAM, em que a solução de $U = MU$ também é solução de

$$\min_{M \in \mathbb{R}^{m \times n}} \|U - MU\|_F^2.$$

Analisando a matriz M , sabemos

A OLAM é uma SPAM

Vamos retomar o caso da OLAM, em que a solução de $U = MU$ também é solução de

$$\min_{M \in \mathbb{R}^{m \times n}} \|U - MU\|_F^2.$$

Analisando a matriz M , sabemos

- $M_O^* = U(U^T U)^{-1} U^T$

A OLAM é uma SPAM

Vamos retomar o caso da OLAM, em que a solução de $U = MU$ também é solução de

$$\min_{M \in \mathbb{R}^{m \times n}} \|U - MU\|_F^2.$$

Analisando a matriz M , sabemos

- $M_O^* = U(U^T U)^{-1} U^T$
- $\mathcal{M}_O(\mathbf{x}) = M_O^* \mathbf{x} = U(U^T U)^{-1} U^T \mathbf{x}.$

A OLAM é uma SPAM

Vamos retomar o caso da OLAM, em que a solução de $U = MU$ também é solução de

$$\min_{M \in \mathbb{R}^{m \times n}} \|U - MU\|_F^2.$$

Analisando a matriz M , sabemos

- $M_O^* = U(U^T U)^{-1} U^T$
- $\mathcal{M}_O(\mathbf{x}) = M_O^* \mathbf{x} = U(U^T U)^{-1} U^T \mathbf{x}.$

Por outro lado,

- $\mathcal{M}_O(\mathbf{x}) = M_O^* \mathbf{x} = U \underbrace{(U^T U)^{-1} U^T}_{\alpha_O^*} \mathbf{x}$

A OLAM é uma SPAM

É a solução de norma mínima para o vetor dos coeficientes do seguinte problema

$$\alpha_O^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} \|\mathbf{x} - U\alpha\|_2^2. \quad (20)$$

no qual a função objetivo pode ser vista fazendo uso da função distância

$$d_{QM}(\mathbf{v}^k, M(\mathbf{x})) = \sum_{i=1}^n \ell_{QM}(\mathbf{v}^k - M(\mathbf{x})), \quad (21)$$

em que a função perda toma a forma da perda quadrática $\ell_{QM}(x) = x^2$.
Ou seja,

A OLAM é uma SPAM

É a solução de norma mínima para o vetor dos coeficientes do seguinte problema

$$\alpha_O^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} \|\mathbf{x} - U\alpha\|_2^2. \quad (20)$$

no qual a função objetivo pode ser vista fazendo uso da função distância

$$d_{QM}(\mathbf{v}^k, M(\mathbf{x})) = \sum_{i=1}^n \ell_{QM}(\mathbf{v}^k - M(\mathbf{x})), \quad (21)$$

em que a função perda toma a forma da perda quadrática $\ell_{QM}(x) = x^2$.
Ou seja,

$$\alpha_O^*(\mathbf{x}) = \arg \min_{\alpha \in \mathbb{R}^p} d_{QM}(U\alpha, \mathbf{x}).$$

A OLAM é uma SPAM

E combinando essas expressões temos a seguinte relação verdadeira:

$$\mathcal{N}(\mathbf{x}) = M_O^* \mathbf{x} = U[(U^T U)^{-1} U^T \mathbf{x}] = U \alpha_O^*.$$

A OLAM é uma SPAM

E combinando essas expressões temos a seguinte relação verdadeira:

$$\mathcal{N}(\mathbf{x}) = M_O^* \mathbf{x} = U[(U^T U)^{-1} U^T \mathbf{x}] = U \alpha_O^*.$$

Logo, afirmamos que a OLAM é uma SPAM com a função distância sendo a distância euclidiana [Valle,13].

A OLAM é uma SPAM

E combinando essas expressões temos a seguinte relação verdadeira:

$$\mathcal{N}(\mathbf{x}) = M_O^* \mathbf{x} = U[(U^T U)^{-1} U^T \mathbf{x}] = U \alpha_O^*.$$

Logo, afirmamos que a OLAM é uma SPAM com a função distância sendo a distância euclidiana [Valle,13].

Uma das estratégias para melhorar a performance da OLAM é analisar o problema das SPAMs considerando estimadores robustos.

Memória Autoassociativa de Projeção em Subespaço Baseada em Estimadores Robustos

Cada M-estimador é definido colocando hipóteses sobre a função perda.

Memória Autoassociativa de Projeção em Subespaço Baseada em Estimadores Robustos

Cada M-estimador é definido colocando hipóteses sobre a função perda.

Huber define qualquer estimativa α^* , o problema de minimizar a forma

$$\min \sum_{i=1}^n \ell \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right) = \min \sum_{i=1}^n \ell(r_i), \quad (22)$$

e pela equação implícita

$$\sum_{i=1}^n \omega_i \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right) U_{ki}^T = 0, \quad \forall k = 1, \dots, p.$$

definida em termos da *função peso* $\omega_i = \frac{\ell'(r_i)}{r_i}$ [Fox,02].

Uma função ℓ que indica a contribuição do residual r_i na função objetivo, e satisfaz as seguintes propriedades

- (i) não-negatividade, isto é, $\ell(r) \geq 0$;
- (ii) $\ell(0) = 0$;
- (iii) simetria, isto é, $\ell(r) = \ell(-r)$;
- (iv) se $|r_i| > |r_j|$ então $\ell(r_i) \geq \ell(r_j)$, para quaisquer i, j .

É dita uma M-Estimativa.

Uma função ℓ que indica a contribuição do residual r_i na função objetivo, e satisfaz as seguintes propriedades

- (i) não-negatividade, isto é, $\ell(r) \geq 0$;
- (ii) $\ell(0) = 0$;
- (iii) simetria, isto é, $\ell(r) = \ell(-r)$;
- (iv) se $|r_i| > |r_j|$ então $\ell(r_i) \geq \ell(r_j)$, para quaisquer i, j .

É dita uma M-Estimativa.

se a função distância d em (18) for diferenciável seu mínimo é obtido ao tomar as derivadas com respeito às coordenadas de α e igualar à zero.

Uma função ℓ que indica a contribuição do residual r_i na função objetivo, e satisfaz as seguintes propriedades

- (i) não-negatividade, isto é, $\ell(r) \geq 0$;
- (ii) $\ell(0) = 0$;
- (iii) simetria, isto é, $\ell(r) = \ell(-r)$;
- (iv) se $|r_i| > |r_j|$ então $\ell(r_i) \geq \ell(r_j)$, para quaisquer i, j .

É dita uma M-Estimativa.

se a função distância d em (18) for diferenciável seu mínimo é obtido ao tomar as derivadas com respeito às coordenadas de α e igualar à zero.

Neste caso, a estimativa robusta α^* corresponde à solução do problema de quadrados mínimos ponderados

$$\arg \min_{\alpha \in \mathbb{R}^p} \|W(U\alpha - \mathbf{x})\|_2^2 \quad (23)$$

onde $W = \text{diag}(\omega_1, \omega_2, \dots, \omega_n)$ é uma matriz diagonal $n \times n$. Ou seja, resolver (18) é o mesmo que resolver (23).

Uma sugestão de função perda é utilizar ℓ_T , que possui a seguinte função perda associada

$$\omega_T(r) = \begin{cases} (1 - (r/k)^2)^2 & |r| \leq k \\ 0, & |r| > k. \end{cases} \quad (24)$$

Uma sugestão de função perda é utilizar ℓ_T , que possui a seguinte função perda associada

$$\omega_T(r) = \begin{cases} (1 - (r/k)^2)^2 & |r| \leq k \\ 0, & |r| > k. \end{cases} \quad (24)$$

A constante de tuneamento k deve ser selecionada de modo a garantir alta eficiência quando a distribuição é assumida padrão, em particular o valor

$$k = \frac{4.685}{0.6754} \text{mediana}(|r - \text{mediana}(|r|)|),$$

assegura 95% de eficiência quando os erros são normais e também proporciona proteção contra *outliers*.

No contexto das memórias associativas, encontrado α_M^* a memória é obtida através da relação

$$\mathcal{M}_M(\mathbf{x}) = U\alpha_M^*$$

denominada *Memória Autoassociativa de Projeção em Subespaço Baseada no Método Robusto de M-Estimativa* (M-SPAM), introduzida por Valle em 2013.

Exemplo M-SPAM

Retomando os dados do primeiro exemplo, calculamos o vetor dos coeficientes

$$\boldsymbol{\alpha}_M^*(\mathbf{x}) = [1.16 \quad -0.54 \quad -0.26 \quad 0.71]^T,$$

e a saída da M-SPAM

$$\mathcal{M}_M(\mathbf{x}) = [14.11 \quad 2 \quad 9.37 \quad 4 \quad 10 \quad 8]^T.$$

Note que há três valores semelhantes à memória fundamental

$\mathbf{u}^1 = [10 \quad 2 \quad 9 \quad 4 \quad 10 \quad 4]$, e o erro encontrado é $\|\mathbf{u}^1 - \mathcal{M}_M(\mathbf{x})\|_2^2 = 5.75$

O S-estimador introduzido por Rousseeuw e Yohai em 1984 é baseado no método do M-Estimador de Huber, entretanto a abordagem utilizada foi de utilizar uma escala de residuais com o objetivo de melhorar o ponto de quebra do estimador.

Quando $\ell : \mathbb{R} \rightarrow [0, \infty)$ que indica a contribuição do residual r_i na função objetivo, e satisfaz as seguintes propriedades

- (i) ℓ é simétrica, continuamente diferenciável e $\ell(0) = 0$;
- (ii) existe $c > 0$ tal que ℓ é estritamente crescente em $[0, c]$ e constante em $[c, \infty)$.

o **S-Estimador** α_S^* é definido por

$$\alpha_S^* = \arg \min_{\alpha \in \mathbb{R}^n} \sigma(r_1(\alpha), r_2(\alpha), \dots, r_n(\alpha)) \quad (25)$$

e a estimativa final ponderada é

$$\hat{\sigma} = \sigma(r_1(\alpha_S^*), r_2(\alpha_S^*), \dots, r_n(\alpha_S^*)). \quad (26)$$

Podemos interpretar $\sigma(\mathbf{r}(\alpha))$ como a medida do tamanho dos valores absolutos do resíduo. Assim uma S-estimativa pode ser vista como a minimização de $\sigma(\mathbf{r}(\alpha))$ para diferentes escalas de estimativa σ .

e a estimativa final ponderada é

$$\hat{\boldsymbol{\sigma}} = \sigma(r_1(\boldsymbol{\alpha}_S^*), r_2(\boldsymbol{\alpha}_S^*), \dots, r_n(\boldsymbol{\alpha}_S^*)). \quad (26)$$

Podemos interpretar $\sigma(\mathbf{r}(\boldsymbol{\alpha}))$ como a medida do tamanho dos valores absolutos do resíduo. Assim uma S-estimativa pode ser vista como a minimização de $\sigma(\mathbf{r}(\boldsymbol{\alpha}))$ para diferentes escalas de estimativa σ .

Ao final, o coeficiente estimado é $\boldsymbol{\alpha}_S^*$ e

$$\mathcal{M}_S(\mathbf{x}) = U\boldsymbol{\alpha}_S^*$$

é a chamada *Memória Autoassociativa de Projeção em Subespaço Baseada na S-Estimativa* (S-SPAM).

Exemplo S-SPAM

Retomando os dados do primeiro exemplo, calculamos o vetor dos coeficientes

$$\boldsymbol{\alpha}_S^*(\mathbf{x}) = [1.08 \quad -0.08 \quad 0.10 \quad -0.28]^T,$$

e a saída da S-SPAM

$$\mathcal{M}_S(\mathbf{x}) = [8.95 \quad 1.08 \quad 9 \quad 4 \quad 10 \quad 2.08]^T.$$

Obtemos valores parecidos ao da M-SPAM. Note que há três valores semelhantes à memória fundamental $\mathbf{u}^1 = [10 \quad 2 \quad 9 \quad 4 \quad 10 \quad 4]$, e o erro encontrado é $\|\mathbf{u}^1 - \mathcal{M}_S(\mathbf{x})\|_2^2 = 2.36$ menor que o encontrado pela M-SPAM.

MM-Estimativa

Esse MM-estimador introduzido por Yohai em 1987 com o objetivo de melhorar o ponto de quebra de um M-Estimador utilizando-se de uma estimativa de escala e de uma M-estimativa.

Esse MM-estimador introduzido por Yohai em 1987 com o objetivo de melhorar o ponto de quebra de um M-Estimador utilizando-se de uma estimativa de escala e de uma M-estimativa.

O **MM-Estimador** é dado em três estágios:

(1º estágio) Toma uma estimativa α_0 com alto ponto de quebra.

(2º estágio) Calcula os resíduos

$$r_i(\alpha) = y_i - \sum_{j=1}^p U_{ij}\alpha_j, \quad \forall i = 1, \dots, n.$$

e calcula uma S-estimativa $\hat{\sigma} = \sigma(\mathbf{r}(\alpha))$ utilizando uma função ℓ_A como função perda.

(3º estágio) Seja l_A e l_B satisfazendo

- (i) $l_i(0) = 0$, l_i simétrica e l_i contínua;
- (ii) $0 < u < v$ implica $l_i(u) \leq l_i(v)$;
- (iii) seja $a = \sup l_i(u)$ então $0 < a < \infty$;
- (iv) se $l_i(u) < a$ e $0 < u < v$ então $l_i(u) < l_i(v)$;

para $i = A, B$ e em particular,

(v) $l_B(u) \leq l_A(u)$ e $\sup l_B(u) = \sup l_A(u) = a$.

Denote $\omega = l'_B$ então a **MM-Estimativa** α_{MM}^* é definida como a solução de

$$\sum_{i=1}^n \omega \left(\frac{r_i(\alpha)}{\sigma(\mathbf{r}(\alpha))} \right) U_{ik} = 0, \quad \forall k = 1, \dots, p$$

o qual verifica $d(\alpha_{MM}^*) \leq d(\alpha_S^*)$ onde

$$d(\alpha) = \sum_{i=1}^n l_B \left(\frac{r_i(\alpha)}{\sigma(\mathbf{r}(\alpha))} \right).$$

Encontrado a estimativa α_{MM}^* podemos recuperar a memória através da expressão

$$\mathcal{M}_{MM}(\mathbf{x}) = U\alpha_{MM}^*$$

chamada Memória Autoassociativa de Projeção em Subespaço baseada no MM-Estimador (MM-SPAM).

Exemplo MM-SPAM

Retomando os dados do primeiro exemplo, calculamos o vetor dos coeficientes

$$\boldsymbol{\alpha}_{MM}^*(\mathbf{x}) = [0.02 \quad 1.95 \quad -0.13 \quad -0.11]^T,$$

e a saída da MM-SPAM

$$\mathcal{M}_{MM}(\mathbf{x}) = [4.35 \quad 5.95 \quad 7.13 \quad 2.31 \quad 9.11 \quad 3.71]^T.$$

Note que não há valores semelhantes à memória fundamental $\mathbf{u}^1 = [10 \quad 2 \quad 9 \quad 4 \quad 10 \quad 4]$, e o erro encontrado é $\|\mathbf{u}^1 - \mathcal{M}_{MM}(\mathbf{x})\|_2^2 = 7.39$. Este erro é maior que o encontrado pela S-SPAM, mas menor que o encontrado pela M-SPAM.

Memória Autoassociativa de Projeção em Subespaço Baseada em Estimadores Robustos

Apesar dos estimadores de regressão robusto serem resistentes à outliers, escolher estimadores levando em conta o ponto de quebra, pode não ser vantajoso se tratando de memórias associativas, já que em geral a matriz das memórias fundamentais não deve ser corrompida.

Memória Autoassociativa de Projeção em Subespaço Baseada em Estimadores Robustos

Apesar dos estimadores de regressão robusto serem resistentes à outliers, escolher estimadores levando em conta o ponto de quebra, pode não ser vantajoso se tratando de memórias associativas, já que em geral a matriz das memórias fundamentais não deve ser corrompida.

Outra abordagem para o problema surge quando removemos a hipótese de que d deve ser diferenciável, assim d pode ser uma aplicação que permite erros.

ε -Regressão de Máquina de Vetor de Suporte

Uma ε -Regressão de Máquina de Vetor de Suporte (ε -SVR) para o caso da regressão multilinear é construída utilizando a função perda ε -insensível ℓ_ε que ignora os erros entre a estimativa e o verdadeiro valor, menores que ε .

ε -Regressão de Máquina de Vetor de Suporte

Uma ε -Regressão de Máquina de Vetor de Suporte (ε -SVR) para o caso da regressão multilinear é construída utilizando a função perda ε -insensível ℓ_ε que ignora os erros entre a estimativa e o verdadeiro valor, menores que ε .

Em particular, a curva estimada f deve ser o mais plana possível, o que é descrito pedindo que o vetor dos coeficientes α seja próximo de zero.

Essa particularidade é motivada do fato de que os dados de treino são assumidos gerados pela mesma relação de fundo.

O objetivo da ε -SVR é obter um erro pequeno ao minimizar o risco funcional

$$R_{emp}[d_\varepsilon] = \frac{1}{2} \|\alpha\|^2 + \frac{C}{n} d_\varepsilon(\mathbf{x}, U\alpha)$$

onde a função distância

$$d_\varepsilon(\mathbf{x}, U\alpha) := \sum_{i=1}^n \ell_\varepsilon \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right)$$

mede a perda ε -insensível dos erros de treino e $C > 0$ é a constante que determina a associação entre a complexidade do modelo (a planitude da curva) e a quantidade de desvios maiores que ε que são tolerados [haykin09,smola08,vapnik99b].

O objetivo da ε -SVR é obter um erro pequeno ao minimizar o risco funcional

$$R_{emp}[d_\varepsilon] = \frac{1}{2} \|\alpha\|^2 + \frac{C}{n} d_\varepsilon(\mathbf{x}, U\alpha)$$

onde a função distância

$$d_\varepsilon(\mathbf{x}, U\alpha) := \sum_{i=1}^n \ell_\varepsilon \left(x_i - \sum_{j=1}^p U_{ij} \alpha_j \right)$$

mede a perda ε -insensível dos erros de treino e $C > 0$ é a constante que determina a associação entre a complexidade do modelo (a planitude da curva) e a quantidade de desvios maiores que ε que são tolerados [haykin09,smola08,vapnik99b].

Desse modo, se C é um valor alto então o objetivo é minimizar a perda, mas se C assume um valor baixo então aprecia-se a planitude da curva.

Formalmente, a ε -SVR realiza uma estimativa utilizando uma função perda ε -insensível e tenta diminuir a complexidade do modelo minimizando $\|\alpha\|^2$.

Formalmente, a ε -SVR realiza uma estimativa utilizando uma função perda ε -insensível e tenta diminuir a complexidade do modelo minimizando $\|\alpha\|^2$.

Para isso, seja $\xi_i^+, \xi_i^- \geq 0$, $i = 1, \dots, n$ denominadas *variáveis de folga*, que penalizam a distância entre as componentes de \mathbf{x} e $U\alpha$. Ou seja,

$$d_\varepsilon(\mathbf{x}, U\alpha) \leq \sum_{i=1}^n (\xi_i^+ + \xi_i^-)$$

logo as restrições sobre as variáveis de folga

$$\begin{aligned} x_i - \sum_{j=1}^p U_{ij}\alpha_j &\leq \varepsilon + \xi_i^+ \\ \sum_{j=1}^p U_{ij}\alpha_j - x_i &\leq \varepsilon + \xi_i^-, \quad \forall i = 1, \dots, n \end{aligned} \tag{27}$$

Formalmente, a ε -SVR realiza uma estimativa utilizando uma função perda ε -insensível e tenta diminuir a complexidade do modelo minimizando $\|\alpha\|^2$.

Para isso, seja $\xi_i^+, \xi_i^- \geq 0$, $i = 1, \dots, n$ denominadas *variáveis de folga*, que penalizam a distância entre as componentes de \mathbf{x} e $U\alpha$. Ou seja,

$$d_\varepsilon(\mathbf{x}, U\alpha) \leq \sum_{i=1}^n (\xi_i^+ + \xi_i^-)$$

logo as restrições sobre as variáveis de folga

$$\begin{aligned} x_i - \sum_{j=1}^p U_{ij}\alpha_j &\leq \varepsilon + \xi_i^+ \\ \sum_{j=1}^p U_{ij}\alpha_j - x_i &\leq \varepsilon + \xi_i^-, \quad \forall i = 1, \dots, n \end{aligned} \tag{27}$$

A estimativa ε -SVR do vetor α é obtida como solução do seguinte problema de otimização

$$\begin{aligned}
&\text{minimizar} && \frac{1}{2} \sum_{j=1}^p \alpha_j^2 + \sum_{i=1}^n (\xi_i^+ + \xi_i^-) \\
&\text{sujeito à} && x_i - \sum_{j=1}^p U_{ij} \alpha_j \leq \varepsilon + \xi_i^+ \\
&&& \sum_{j=1}^p U_{ij} \alpha_j - x_i \leq \varepsilon + \xi_i^- \\
&&& \xi_i^+, \xi_i^- \geq 0, \quad \forall i = 1, \dots, n
\end{aligned} \tag{28}$$

A resolução de uma estimativa ε -SVR, tem seu ótimo dado por α_ε^* . Podemos então calcular a memória $\mathcal{M}_\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}^n$ denominada *Memória Autoassociativa de Projeção em Subespaço Baseada na ε -Regressão de Vetor de Suporte* (ε -SVR SPAM), através da expressão

$$\mathcal{M}_\varepsilon(\mathbf{x}) = U\alpha_\varepsilon^*. \tag{29}$$

Por outro lado, a saída da ε -SVR SPAM não satisfaz a propriedade de ser mínima, ou seja, a (29) não satisfaz

$$d_\varepsilon(\mathbf{x}, \mathcal{M}_\varepsilon(\mathbf{x})) \leq d_\varepsilon(\mathbf{x}, U\alpha), \quad \forall \alpha \in \mathbb{R}^p.$$

Já que o termo $\|\alpha\|_2^2$ em (28) pode não representar o mínimo, logo \mathcal{M}_ε não é uma SPAM.

Por outro lado, a saída da ε -SVR SPAM não satisfaz a propriedade de ser mínima, ou seja, a (29) não satisfaz

$$d_\varepsilon(\mathbf{x}, \mathcal{M}_\varepsilon(\mathbf{x})) \leq d_\varepsilon(\mathbf{x}, U\alpha), \quad \forall \alpha \in \mathbb{R}^p.$$

Já que o termo $\|\alpha\|_2^2$ em (28) pode não representar o mínimo, logo \mathcal{M}_ε não é uma SPAM.

Podemos definir uma SPAM substituindo (28) pela seguinte dupla de programas de otimização

$$\begin{aligned}
\min \quad & \frac{1}{2} \sum_{j=1}^p \alpha_j^2, \\
\text{s. à} \quad & \sum_{i=1}^n (\xi_i^+ + \xi_i^-) \leq \mathcal{E}^*, \\
& x_i - \sum_{j=1}^p U_{ij} \alpha_j \leq \varepsilon + \xi_i^+, \\
& \sum_{j=1}^p U_{ij} \alpha_j - x_i \leq \varepsilon + \xi_i^-, \\
& \xi_i^+, \xi_i^- \geq 0, \quad \forall i = 1, \dots, n
\end{aligned}$$

$$\begin{aligned}
\min \quad & \sum_{i=1}^n (\xi_i^+ + \xi_i^-) \\
\text{s. à} \quad & x_i - \sum_{j=1}^p U_{ij} \alpha_j \leq \varepsilon + \xi_i^+ \\
& \sum_{j=1}^p U_{ij} \alpha_j - x_i \leq \varepsilon + \xi_i^- \\
& \xi_i^+, \xi_i^- \geq 0, \quad \forall i = 1, \dots, n.
\end{aligned}$$

(30)

sendo \mathcal{E}^* é a solução do programa linear (38). Dessa forma, a solução $\alpha'(\mathbf{x})$ da dupla de problemas de otimização necessariamente minimiza $d_\varepsilon(\mathbf{x}, U\alpha)$, $\forall \alpha \in \mathbb{R}^p$.

Apesar da abordagem utilizada pela dupla de problemas de otimização, preferimos (28) por ser baseado na abordagem padrão da ε -SVR. E também, é possível encontrar $C \geq 0$ capaz de obter α_ε^* solução de (28) muito similar à solução α' da dupla de problemas de otimização.

Apesar da abordagem utilizada pela dupla de problemas de otimização, preferimos (28) por ser baseado na abordagem padrão da ε -SVR. E também, é possível encontrar $C \geq 0$ capaz de obter α_ε^* solução de (28) muito similar à solução α' da dupla de problemas de otimização.

Em muitas situações, o programa quadrático (28) é solucionado através da sua forma dual, obtida utilizando o método de Lagrange.

Logo o programa dual associado à (28) é o seguinte

$$\begin{aligned}
 \text{maximizar} \quad & -\frac{1}{2} \sum_{j=1}^p \left[\sum_{i=1}^n (\lambda_i^+ - \lambda_i^-) U_{ij} \right]^2 \\
 & - \sum_{i=1}^n (\lambda_i^+ + \lambda_i^-) \varepsilon + \sum_{i=1}^n (\lambda_i^+ - \lambda_i^-) x_i \\
 \text{sujeito à} \quad & 0 \leq \lambda_k^+ \leq C \\
 & 0 \leq \lambda_k^- \leq C \quad \forall k = 1, \dots, n.
 \end{aligned} \tag{31}$$

A solução do problema de otimização quadrática dual é em termos dos multiplicadores de Lagrange λ^+ , λ^- e não do vetor dos coeficientes α desejada.

A partir do método de multiplicadores de Lagrange é possível obter a seguinte estimativa para α

$$\alpha_k = \sum_{i=1}^n (\lambda_i^+ - \lambda_i^-) U_{ik}, \quad \forall k = 1, \dots, p.$$

é denominada expansão de vetor de suporte.

A partir do método de multiplicadores de Lagrange é possível obter a seguinte estimativa para α

$$\alpha_k = \sum_{i=1}^n (\lambda_i^+ - \lambda_i^-) U_{ik}, \quad \forall k = 1, \dots, p.$$

é denominada expansão de vetor de suporte.

Dualmente, $\mathcal{M}_\varepsilon^*$ denota a ε -SVR SPAM baseada na formulação dual. Combinando (29) e α , a ε -SVR SPAM $\mathcal{M}_\varepsilon^* : \mathbb{R}^n \rightarrow \mathbb{R}^n$ definida por

$$\mathcal{M}_\varepsilon^*(\mathbf{x}) = W(\check{\lambda}^+(\mathbf{x}) - \check{\lambda}^-(\mathbf{x})), \quad \forall \mathbf{x} \in \mathbb{R}^n, \quad (32)$$

onde $\check{\lambda}^+(\mathbf{x})$ e $\check{\lambda}^-(\mathbf{x})$ denotam a solução (31) e $W = UU^T$ é a matriz peso sináptico produzida pela regra do produto externo [Haykin,09].

Exemplo ε -SVR SPAM

Retomando os dados do primeiro exemplo, fixando $C = 10$ e $\varepsilon = 0.05$, calculamos o vetor dos coeficientes utilizando a formulação primal

$$\alpha_\varepsilon(\mathbf{x}) = [0.99 \quad 0.01 \quad -0.01 \quad 0.02]^T,$$

com as variáveis de folga $\check{\xi}^+ - \check{\xi}^-$ que aproximam a diferença $\mathbf{x} - \mathcal{M}_\varepsilon(\mathbf{x})$.

$$\check{\xi}^+(\mathbf{x}) = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 3.84]^T,$$

$$\check{\xi}^-(\mathbf{x}) = [9.96 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T.$$

Assim podemos calcular a saída

$$\mathcal{M}_\varepsilon(\mathbf{x}) = [10.01 \quad 2.05 \quad 8.95 \quad 3.95 \quad 9.95 \quad 4.11]^T$$

Por outro lado, calculamos os multiplicadores de Lagrange, utilizando a formulação dual

$$\check{\lambda}^+(\mathbf{x}) = [0.00 \ 0.00 \ 0.98 \ 0.75 \ 0.26 \ 1.67]^T,$$

$$\check{\lambda}^-(\mathbf{x}) = [1.67 \ 1.66 \ 0.00 \ 0.00 \ 0.00 \ 0.00]^T.$$

e a recordação

$$\mathcal{M}_\varepsilon^*(\mathbf{x}) = \underbrace{\begin{bmatrix} 209 & 134 & 150 & 132 & 163 & 177 \\ 134 & 137 & 88 & 121 & 94 & 152 \\ 150 & 88 & 122 & 92 & 135 & 110 \\ 132 & 121 & 92 & 117 & 98 & 134 \\ 163 & 94 & 135 & 98 & 150 & 117 \\ 177 & 152 & 110 & 134 & 117 & 189 \end{bmatrix}}_W \underbrace{\begin{bmatrix} -1.67 \\ -1.66 \\ 0.98 \\ 0.75 \\ 0.26 \\ 1.67 \end{bmatrix}}_{\check{\lambda}^+(\mathbf{x}) - \check{\lambda}^-(\mathbf{x})}$$

$$= [10.01 \ 2.05 \ 8.95 \ 3.95 \ 9.95 \ 4.11]^T,$$

muito semelhante à da formulação dual.

Por outro lado, calculamos os multiplicadores de Lagrange, utilizando a formulação dual

$$\check{\lambda}^+(\mathbf{x}) = [0.00 \ 0.00 \ 0.98 \ 0.75 \ 0.26 \ 1.67]^T,$$

$$\check{\lambda}^-(\mathbf{x}) = [1.67 \ 1.66 \ 0.00 \ 0.00 \ 0.00 \ 0.00]^T.$$

e a recordação

$$\mathcal{M}_\varepsilon^*(\mathbf{x}) = \underbrace{\begin{bmatrix} 209 & 134 & 150 & 132 & 163 & 177 \\ 134 & 137 & 88 & 121 & 94 & 152 \\ 150 & 88 & 122 & 92 & 135 & 110 \\ 132 & 121 & 92 & 117 & 98 & 134 \\ 163 & 94 & 135 & 98 & 150 & 117 \\ 177 & 152 & 110 & 134 & 117 & 189 \end{bmatrix}}_W \underbrace{\begin{bmatrix} -1.67 \\ -1.66 \\ 0.98 \\ 0.75 \\ 0.26 \\ 1.67 \end{bmatrix}}_{\check{\lambda}^+(\mathbf{x}) - \check{\lambda}^-(\mathbf{x})}$$

$$= [10.01 \ 2.05 \ 8.95 \ 3.95 \ 9.95 \ 4.11]^T,$$

muito semelhante à da formulação dual. O erro para ambos os casos é o mesmo, $\|\mathbf{u}^1 - \mathcal{M}_\varepsilon(\mathbf{x})\|_2 = 0.15$, que foi o menor erro encontrado nos modelos expostos.

- Apresentamos a classe das SPAMs;
- Apresentamos modelos de SPAMs baseados no S- e MM-Estimador robusto;
- Apresentamos modelos de SPAMs baseados na ε -Regressão de Máquina de Vetor de Suporte;
- Pretendemos investigar o comportamento do ε -SVR SPAM comparado com outros modelos de memórias.

Obrigada!