Principal component regression (PCR)

- met. extrakcie prem. : aj ako vstupy do predikčných modelov
  ↳ PCA pre lin. reg. : PCR ..., regresia na hlavných komponentoch

dáta : y, X

1.) PCA : vycentrujeme X (+ vyškálujeme) $\Rightarrow$ PCA : $Z = XQ$ ($n\times q$ $n\times q$ $q\times q$)
   - zoberieme prvých q hl. komp. (stĺpcov Z) $\Rightarrow \tilde{Z} = \begin{bmatrix} 1 & | & | & & | \\ 1 & z_1 & \cdots & z_q \\ 1 & | & | & & | \end{bmatrix}$

2.) lin. reg. : $y = \tilde{Z}\beta + \varepsilon$ (*)
   → robíme s (*)

- môže zabrániť pretrénovaniu (chránime sa voči vys. šumu)
  → nepozeráme sa pri PCA na y ⇒ nemusí dobre fungovať
  PCR: predpokladá, že smery X s najv. rozptylom sú najviac informatívne pre y
  - voľba q : krosval., val.
  - výpočet (*) : jednoduchý, lebo stĺpce Z sú ortogonálne
  - typicky X štandardizujeme pred PCA
  - vtáčik s Ridge : tá zmenšuje β viac pre nesprávne PC
    : PCR úplne vynuluje β od niektorého PC

Hitters
- PC1 : vysoké – aktívni a starí
- PC2 : vysoké – menej aktívni a starí
- SV: starí, IV: aktívni

Autoencoders (Autoassociative neural networks)
- autoasociatívne neurónové siete

Neurónové sieťe (NS)

lin. reg. : $y_i = x_i^T\beta + \varepsilon_i$
$\hat{y}_i = x_i^T\hat{\beta}$
pre nové x : $\hat{y}(x) = x^T\hat{\beta}$
$\hat{\beta}$: $\min_\beta \sum_{i=1}^n (y_i - x_i^T\beta)^2$

→ Machine learning pohľad:
1. štruktúra modelu;
   vstup $x \in \mathbb{R}^p$ → výstup : $\hat{y}(x) = x^T\beta$ (pre dané β)
2. fitovanie :
   dáta : $x_1, \dots, x_n$ → $\hat{y}_i = x_i^T\beta$
   $y_1, \dots, y_n$
   $\Rightarrow \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - x_i^T\beta)^2 \to \min_\beta$
   $\Rightarrow \beta^*$: natrénované parametre (váhy)

VNS : komplikovanejší 1.
↳ pomocou $x_1, \dots, x_n$ predikujeme y_i

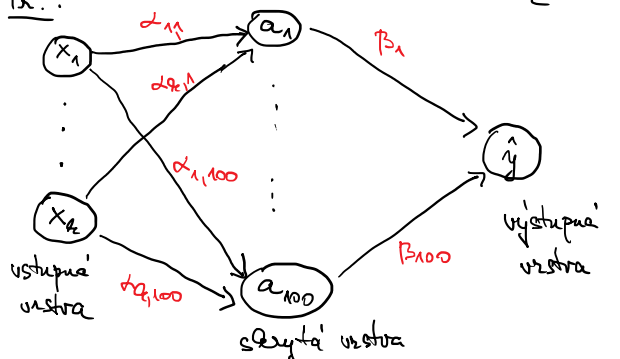
1. model : $\hat{y} = f_\theta(x)$
   ↑ parametre (váhy) NS

aktivačná fcia (nie nutne)

1. model : $\hat{y} = f_\theta(x)$
↑ parametre (váhy) NS

— ako neuróny v mozgu

Pr.:



pre $y \in \mathbb{R}$

aktivačná fcia (nelineárna)

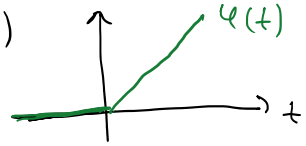
$a_1 = \varphi_1(\alpha_{1,0} + \alpha_{1,1}x_1 + \ldots + \alpha_{p,1}x_p)$

$\vdots$

$a_{100} = \varphi_1(\alpha_{100,0} + \alpha_{1,100}x_1 + \ldots + \alpha_{p,100}x_p)$

$f_\theta(x) = \hat{y} = \varphi_2(\beta_0 + \beta_1 a_1 + \ldots + \beta_{100} a_{100})$

napr. $\varphi(t) = \max(0, t)$ ReLU



$\theta = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$

— sily: ako neuróny potrebujú informácie

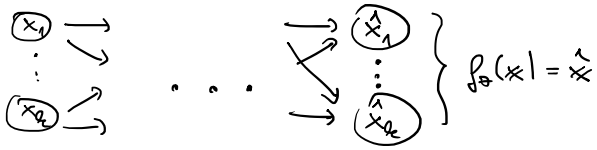
2. učiaca : napr. $x_1, \ldots, x_n \in \mathbb{R}^p$

$y_1, \ldots, y_n \in \mathbb{R}$ $\Rightarrow \theta^* \in \operatorname{argmin}_\theta \sum_{i=1}^{n} (y_i - f_\theta(x_i))^2$

odhady: $f_\theta(x_1), \ldots, f_\theta(x_n)$

autoencoders :

1. pomocou $x \in \mathbb{R}^p$ modeluje znovu $x \in \mathbb{R}^p$
→ "auto", "autoassociative"



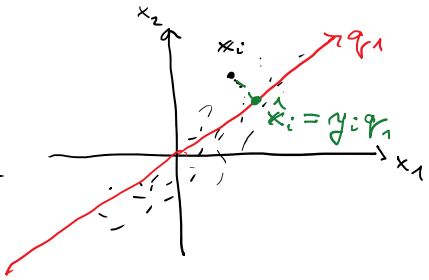
2. $\hat{x}_i$ blízke x_i

$\Rightarrow$ chyba $= \sum_{i=1}^{n} \| x_i - f_\theta(x_i) \|^2 \to \min_\theta : \theta^*$

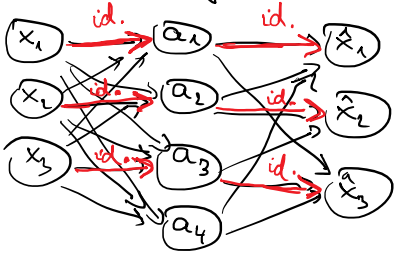
$x_i - f_\theta(x_i) \in \mathbb{R}^p$

- podobne PCA: vstup : $x \in \mathbb{R}^p$

: výstup : $y \in \mathbb{R}^q$ ... $\hat{x}$ blízke x, a y sú jeho súradnice



Pr.: $x \in \mathbb{R}^3 \mapsto f_\theta(x) \in \mathbb{R}^3$, 3 vrstvy



optimálne : $\hat{x}_1 = x_1$
$\hat{x}_2 = x_2$
$\hat{x}_3 = x_3$

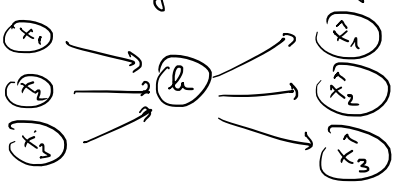
$a_1 = 1 \cdot x_1 + 0 \cdot x_2 + 0 \cdot x_3$
$\hat{x}_1 = 1 \cdot a_1 + 0 \cdot a_2 + 0 \cdot a_3 + 0 a_4$
atď.

→ zbytočné

$\Rightarrow$ cieľ : zabrániť, aby sa naučil identitu

Trik : v skrytej vrstve menej uzlov

Pr.:



- všeob. : skrytá vrstva : $q < p$ uzlov

- všeob.: skrytá vrstva: $q < k$ uzlov
- ak φ - lineárna (identita) $\Rightarrow$: autoencoder $\Leftrightarrow$ PCA

  ↑

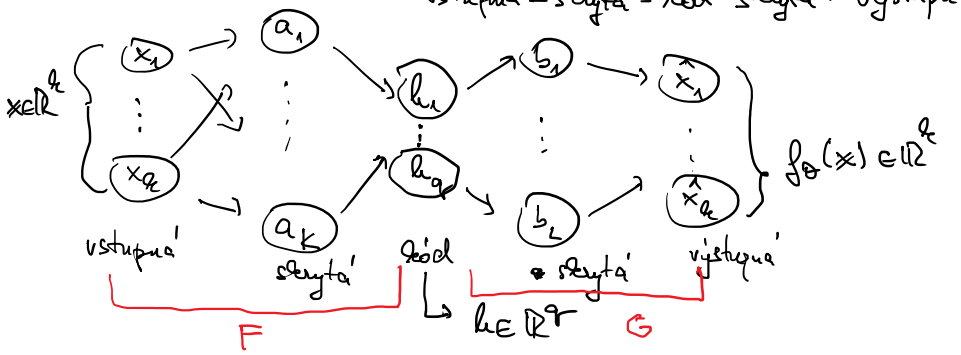
  výstupy $\hat{x}_1, \dots, \hat{x}_n \in \mathbb{R}^k$ ležia v priestore prvých q hl. vektorov S
- ak iba 1 skrytá vrstva: v teórii by sme mali dostať PCA

$\Rightarrow$ autoencoder – nelineárne φ

– typicky aspoň 5 vrstiev: tá v strede; $q < k$ uzlov

↳ kód (code) $\Rightarrow$ "encoder" (zakódovanie, rozkódovanie)

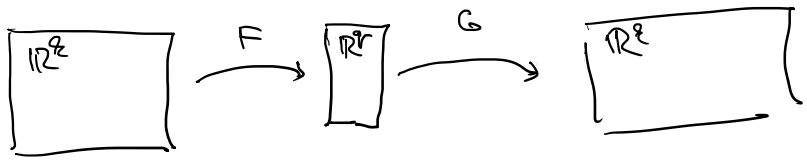
↳ vstupná – skrytá – kód – skrytá – výstupná



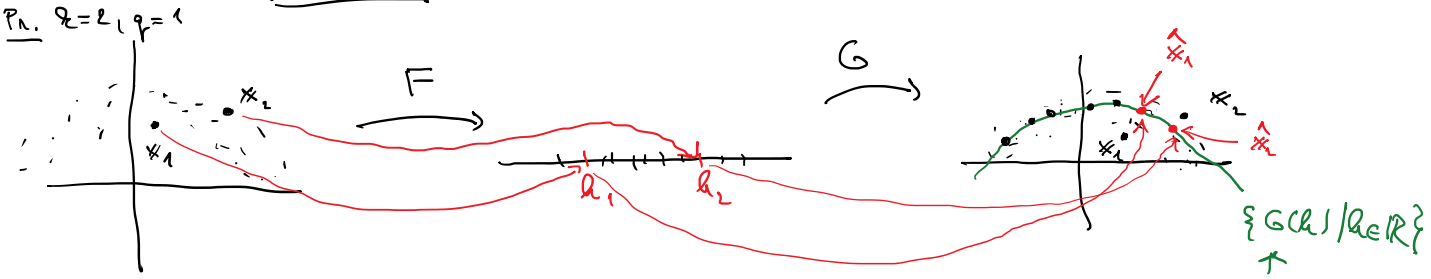
1.) $h = F(x)$ } $f_\theta(x) = G(F(x))$, $F: \mathbb{R}^k \to \mathbb{R}^q$, $G: \mathbb{R}^q \to \mathbb{R}^k$

2.) $f_\theta(x) = G(h)$

schematicky:



Pr. $k=2$, $q=1$



$\{G(h) / h \in \mathbb{R}\}$

↑

priamka, ak je 1-D

- q - rozumná reprezentácia: výstupy kódu: $h_1, \dots h_n \in \mathbb{R}^q$
- autoencoder: "nelineárna PCA" – niekedy sa tak aj nazýva

použitie: vyčistenie

: zabránenie pretrénovaniu

: odšumenie/vyčistenie dát, obrázkov

- ∃ iné varianty

Príp.:

1.) $x_1, \dots, x_k \to h_1, \dots, h_q \to \hat{x}_1, \dots, \hat{x}_k$

2.) 6 – 4 – 2 – 4 – 6

vstup ↑ ↑ ↓ výstup

kód

pre MNIST

784 — sigm. 28 — sig. 2 — sig. 28 — sig. 784

$\varphi(t) =$ "sigmoid" $: \mathbb{R} \to [0,1]$