Detecção de Anomalias
e Sistemas de Recomendação

Prof. Danilo Silva

EEL7514/EEL7513 - Tópico Avançado em Processamento de Sinais:
Introdução ao Aprendizado de Máquina

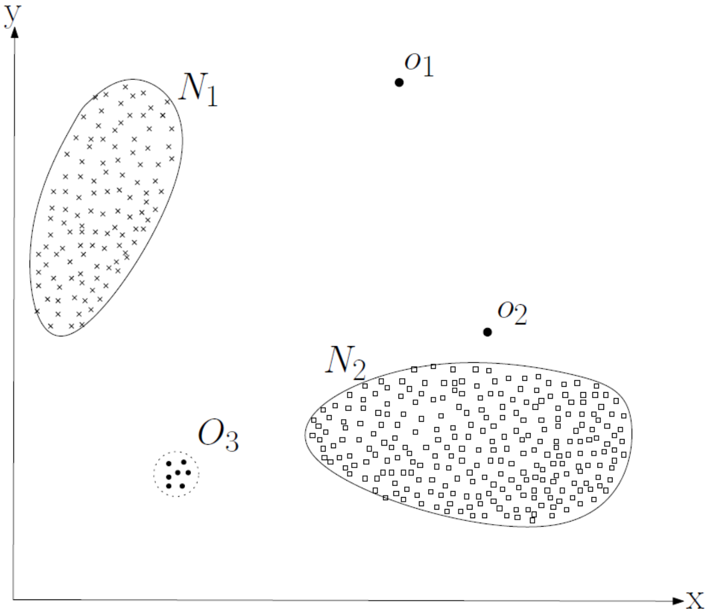
EEL / CTC / UFSC

Detecção de Anomalias

Detecção de Anomalias (ou Outliers)

- Objetivo: detectar casos que fogem ao "normal" (comum / esperado)
- Diferentemente do aprendizado supervisionado, caracteriza-se por haver um número muito pequeno (ou nenhum) de amostras rotuladas como anômalas
- Aplicações:
  - Detecção de intrusos / atividade maliciosa / fraudes
  - Detecção de falhas em sistemas
  - Monitoramento de saúde
- Abordagens:
  - Baseada em classificação (ex: SVM de única classe)
  - Redução de dimensionalidade (ex: PCA, redes neurais auto-replicantes)
  - Vizinhança/clustering
  - Modelamento estatístico (estimação de densidade de probabilidade)

Exemplo



Estimação de Densidade

- Princípio básico: uma amostra é classificada como anômala se possui baixa probabilidade de ocorrência (abaixo de um limiar pré-estabelecido)

$$p(\mathbf{x}) < \epsilon$$

Tipos:

- Estimação paramétrica: assume um modelo específico caracterizado por uma densidade de probabilidade com parâmetros livres a serem ajustados pelos dados (ex: modelo gaussiano)
- Estimação não-paramétrica: não faz hipóteses sobre o modelo, ao invés disso determina a densidade a partir dos dados (ex: histograma)

Modelo Gaussiano (Univariável)

- $n = 1 \implies \mathbf{x} = (x_1) = x \in \mathbb{R}$
- Densidade de probabilidade ($x \sim \mathcal{N}(\mu, \sigma^2)$):

$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

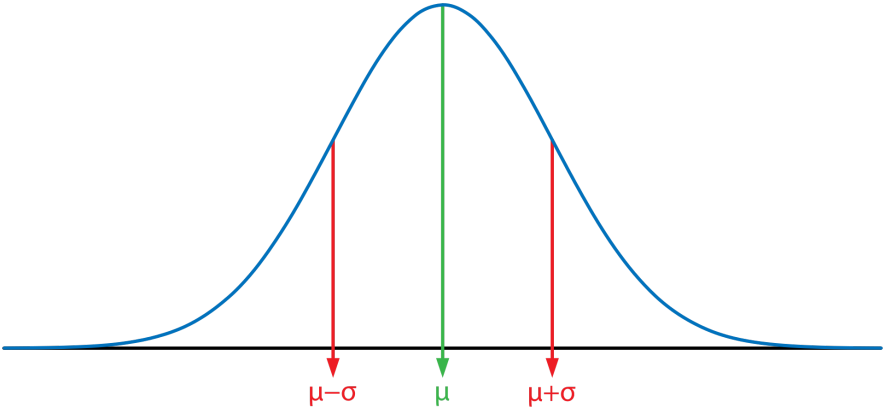
- Estimação de máxima verossimilhança:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}$$

$$\sigma^2 = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu)^2$$

  - Estimação não-enviesada de σ^2: divida por $m - 1$ ao invés de m
    - Obs: irrelevante para m suficientemente grande

Exemplo



Modelo Gaussiano Multivariável

- Densidade de probabilidade ($\mathbf{x} \in \mathbb{R}^n, \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$):

$$p(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

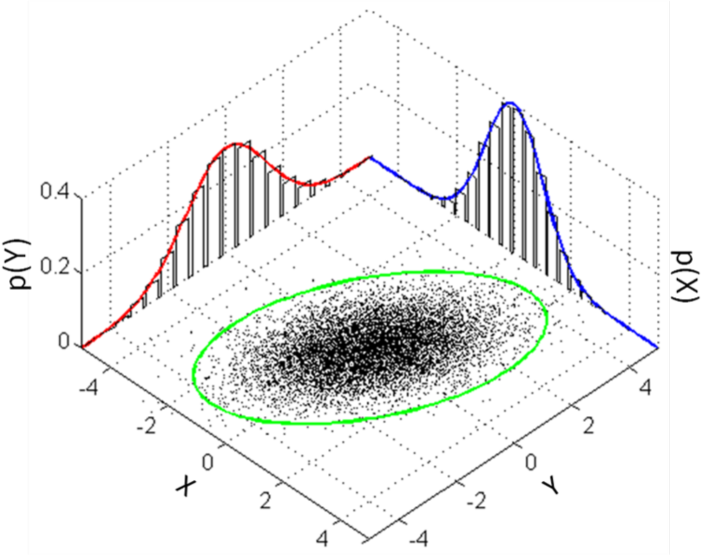
- Estimação de máxima verossimilhança:

$$\boldsymbol{\mu} = \frac{1}{m} \sum_{i=1}^{m} \mathbf{x}^{(i)}$$

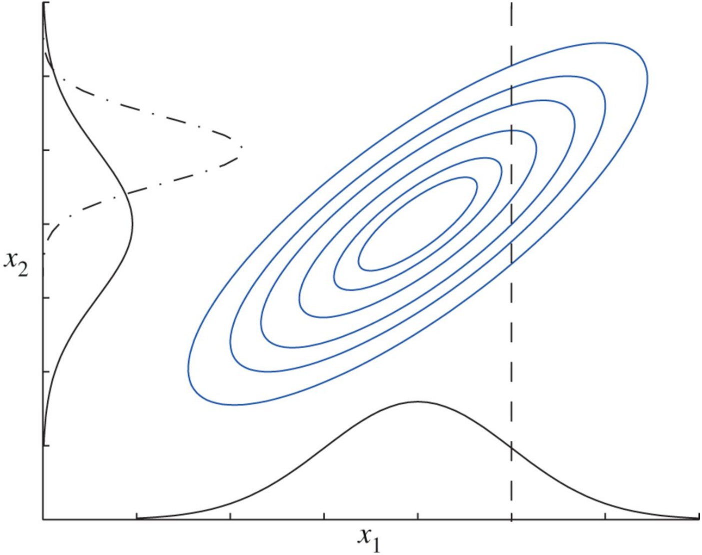
$$\boldsymbol{\Sigma} = \frac{1}{m} \sum_{i=1}^{m} (\mathbf{x}^{(i)} - \boldsymbol{\mu})(\mathbf{x}^{(i)} - \boldsymbol{\mu})^T$$

  - Obs: requer $m > n$ para que $\boldsymbol{\Sigma}$ seja inversível
- Modelar cada x_j como uma variável gaussiana independente equivalente a modelar $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ onde $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1^2, \ldots, \sigma_n^2)$
  - Nesse caso, se houver correlação nos dados ela não será identificada
  - Em compensação, requer menor complexidade e menos dados

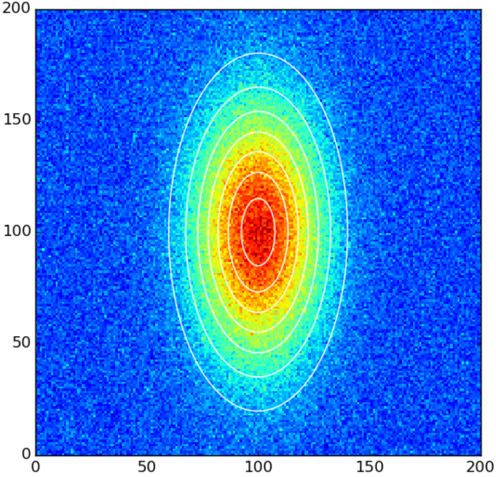
Exemplo



Exemplo



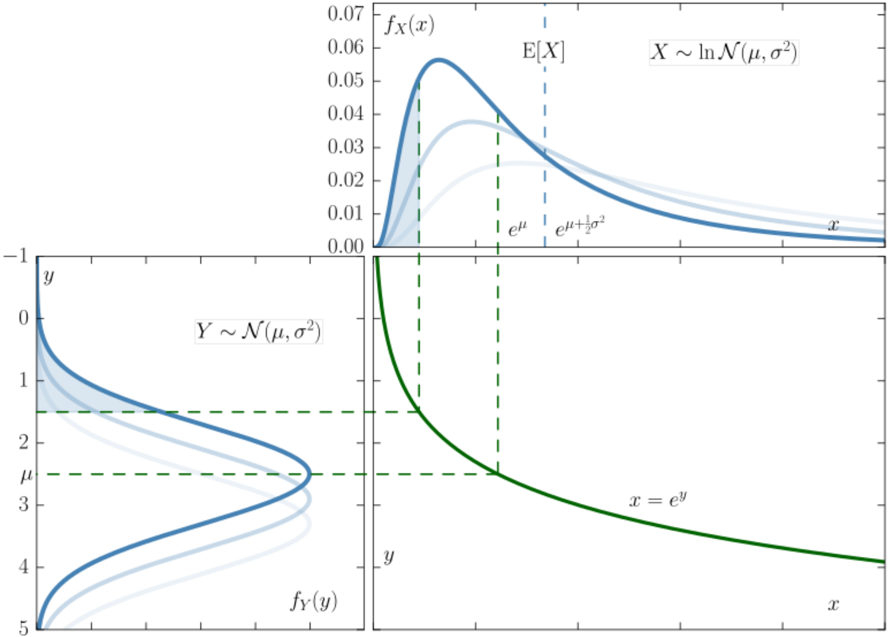
Exemplo



Recomendações

- Visualize os dados antes de modelar
- Mesmo que a distribuição não seja aproximadamente gaussiana, é possível que se torne aproximadamente gaussiana após alguma transformação
  - Ex: X é log-normal $\implies Y = \log(X)$ é gaussiana
- Escolha atributos que possam variar significativamente no caso de uma anomalia
- Ajuste o modelo em amostras normais; se possuir amostras anômalas, guarde-as para validação e/ou teste

Exemplo



0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
$f_X(x)$
E[X]
$X \sim \ln \mathcal{N}(\mu, \sigma^2)$
e^{μ}
$e^{\mu+\frac{1}{2}\sigma^2}$
x
-1
0
1
2
μ
3
4
5
y
$Y \sim \mathcal{N}(\mu, \sigma^2)$
$f_Y(y)$
$x = e^y$
y
x

Avaliação do Modelo

- Amostras rotuladas como anômalas podem ser usadas para avaliação do modelo
- Deve ser usada uma métrica robusta a desbalanceamento das classes (não usar acurácia):
  - Curva ROC (TPR x FPR)
  - Curva Precision-Recall

$$P = \frac{T_p}{T_p + F_p}, \qquad R = \frac{T_p}{T_p + F_n}$$

  - F_1 score (obs: "Positivo" = "Anômalo"):

$$F_1 = 2\frac{PR}{P + R} = \frac{2T_p}{2T_p + F_n + F_p}$$

    - Vantagem de ser um único número

Como Determinar o Limiar?

- Amostras rotuladas como anômalas também podem ser usadas para escolher o limiar ϵ, através de um conjunto de validação
- Escolha ϵ que maximiza o desempenho no conjunto de validação

Sistemas de Recomendação

Sistemas de Recomendação

- Objetivo: a partir de dados sobre um usuário (avaliações, compras passadas, etc), recomendar itens que este usuário tem mais chance de se interessar
  - Também conhecido como “filtragem” (de itens para um dado usuário)
- Exemplos: Amazon, Netflix, Facebook, etc
- Tipos de abordagem:
  - Filtragem de conteúdo (*content-based filtering*): requer uma descrição do conteúdo dos itens (atributos)
  - Filtragem colaborativa (*collaborative filtering*): baseia-se exclusivamente nas avaliações de outros usuários
  - Híbrida

Filtragem de Conteúdo

Filtragem de Conteúdo

- Suponha m itens, n atributos e K usuários
- Os itens são descritos por uma matriz $\mathbf{X} \in \mathbb{R}^{m \times n}$
- As preferências dos usuários pelos itens são dadas por uma matriz $\mathbf{Y} \in \mathbb{R}^{m \times K}$ com entradas faltantes
  - $Y_{i,k} = y_k^{(i)}$ é a avaliação dada ao item i pelo usuário k
- Problema: determinar os valores das entradas faltantes ($y_k^{(i)} =$?)

Exemplo

Filme	x_1 (romance)	x_2 (ação)	Alice	Bruno	Carol	Davi
Star Wars	0.1	0.93	0	5	5	0
Matrix	0.05	0.99	0	5	?	?
X-Men	0	0.95	?	?	4	0
Titanic	0.99	0.05	4	0	0	5
Uma Linda Mulher	0.92	0	?	0	0	5

Filtragem de Conteúdo: Modelo Linear

- Seja $\mathbf{R} \in \{0, 1\}^{m \times K}$ uma matriz que indica as avaliações conhecidas:

$$R_{ik} = \begin{cases} 1, & Y_{ik} = y_k^{(i)} \neq ? \quad \text{(avaliado)} \\ 0, & Y_{ik} = y_k^{(i)} = ? \quad \text{(não-avaliado)} \end{cases}$$

- Suponha $\mathbf{x}^{(i)} = (1, x_1^{(i)}, \ldots, x_n^{(i)})^T$ e $\mathbf{w}_k = (w_{k0}, w_{k1}, \ldots, w_{kn})^T$
- A avaliação do item i pelo usuário k é modelada como

$$\hat{y}_k^{(i)} = \mathbf{w}_k^T \mathbf{x}^{(i)}$$

Função Custo

- Função custo para o usuário k:

$$J(\mathbf{w}_k) = \frac{1}{2} \sum_{i=1}^{m} R_{ik} (\mathbf{w}_k^T \mathbf{x}^{(i)} - y_k^{(i)})^2$$

- Função custo total:

$$J = J(\mathbf{W}) = \frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{m} R_{ik} (\mathbf{w}_k^T \mathbf{x}^{(i)} - y_k^{(i)})^2$$

- Problema de regressão linear ponderada (ou mínimos quadrados ponderados)
  - Pode ser resolvido pela equação normal ou métodos iterativos

Notação Matricial

- Predição: $\hat{\mathbf{Y}}_{:,k} = \mathbf{X}\mathbf{w}_k \quad \Longrightarrow \quad \hat{\mathbf{Y}} = \mathbf{X}\mathbf{W}^T$
- Função custo para o usuário k:

$$
\begin{aligned}
J(\mathbf{w}_k) &= \frac{1}{2}\|(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k}) \odot \mathbf{R}_{:,k}\|^2 \\
&= \frac{1}{2}(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k})^T \operatorname{diag}(\mathbf{R}_{:,k})(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k})
\end{aligned}
$$

- Gradiente:

$$
\begin{aligned}
\frac{\partial J}{\partial \mathbf{w}_k} &= \mathbf{X}^T \operatorname{diag}(\mathbf{R}_{:,k})(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k}) \\
\frac{\partial J}{\partial \mathbf{W}^T} &= \mathbf{X}^T(\mathbf{R} \odot (\mathbf{X}\mathbf{W}^T - \mathbf{Y}))
\end{aligned}
$$

Com Regularização

- Seja $\mathbf{L} = \mathrm{diag}(0, 1, \ldots, 1) \in \mathbb{R}^{(n+1)\times(n+1)}$
- Função custo para o usuário k:

$$
\begin{aligned}
J(\mathbf{w}_k) &= \frac{1}{2}\|(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k}) \odot \mathbf{R}_{:,k}\|^2 + \frac{\lambda}{2}\mathbf{w}_k^T\mathbf{L}\mathbf{w}_k \\
&= \frac{1}{2}(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k})^T \mathrm{diag}(\mathbf{R}_{:,k})(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k}) + \frac{\lambda}{2}\mathbf{w}_k^T\mathbf{L}\mathbf{w}_k
\end{aligned}
$$

- Gradiente:

$$
\begin{aligned}
\frac{\partial J}{\partial \mathbf{w}_k} &= \mathbf{X}^T \mathrm{diag}(\mathbf{R}_{:,k})(\mathbf{X}\mathbf{w}_k - \mathbf{Y}_{:,k}) + \lambda\mathbf{L}\mathbf{w}_k \\
\frac{\partial J}{\partial \mathbf{W}^T} &= \mathbf{X}^T(\mathbf{R} \odot (\mathbf{X}\mathbf{W}^T - \mathbf{Y})) + \lambda\mathbf{L}\mathbf{W}^T
\end{aligned}
$$

Lidando com novos usuários

- Para um usuário que não avaliou nenhum item:
  - O modelo não tem como aprender um vetor de parâmetros adequado:

$$J(\mathbf{w}_k) = \frac{\lambda}{2} \sum_{j=1}^{n} w_{kj}^2 \qquad \Longrightarrow \qquad \mathbf{w}_k = \begin{bmatrix} w_{k0} & 0 & \cdots & 0 \end{bmatrix}$$

  - Desempenho pior do que prever a média de avaliações de cada item
- Solução: a matriz $\mathbf{Y}$ deve ser previamente centralizada pela média de avaliações dos itens

$$\boldsymbol{\mu} = \begin{bmatrix} \mu^{(1)} & \cdots & \mu^{(m)} \end{bmatrix}^T, \qquad \mu^{(i)} = \frac{\sum_{k=1}^{K} R_{ik} Y_{ik}}{\sum_{k=1}^{K} R_{ik}}$$

$$\mathbf{Y}' = \mathbf{Y} - \boldsymbol{\mu}, \qquad \hat{\mathbf{Y}}' = \mathbf{X}\mathbf{W}^T, \qquad \hat{\mathbf{Y}} = \hat{\mathbf{Y}}' + \boldsymbol{\mu}$$

Limitações

Limitações do modelo linear para filtragem de conteúdo:

- Modelo linear
  - Obs: problema de regressão ponderada
- Filtragem de conteúdo
  - Como determinar $\mathbf{X}$?

Filtragem Colaborativa

Filtragem Colaborativa

- Suponha m itens e K usuários
- As preferências dos usuários pelos itens são dadas por uma matriz $\mathbf{Y} \in \mathbb{R}^{m \times K}$ com entradas faltantes
  - $Y_{i,k} = y_k^{(i)}$ é a avaliação dada ao item i pelo usuário k
- Problema: determinar os valores das entradas faltantes ($y_k^{(i)} = ?$)

Exemplo

Filme	Alice	Bruno	Carol	Davi
Star Wars	0	5	5	0
Matrix	0	5	?	?
X-Men	?	?	4	0
Titanic	4	0	0	5
Uma Linda Mulher	?	0	0	5

Filtragem Colaborativa: Tipos de abordagem

- Baseada em memória: guarda toda a matriz de avaliações e realiza predições baseadas em similaridade de avaliações
  - Considera como vizinhos usuários que possuem preferências semelhantes
  - Recomenda a um usuário itens bem avaliados pelos seus vizinhos
- Baseada em modelo: ajusta aos dados um modelo que possui variáveis latentes (ocultas)
  - Assume que itens e usuários podem ser representados por vetores em um espaço de dimensão n pequena
  - Recomenda a um usuário os itens mais próximos a ele neste espaço latente

Fatoração Matricial de Baixo Posto

Low-Rank Matrix Factorization

- Objetivo: encontrar matrizes $\mathbf{X} \in \mathbb{R}^{m \times (n+1)}$ e $\mathbf{W} \in \mathbb{R}^{K \times (n+1)}$ tais que

$$\mathbf{X}\mathbf{W}^T \approx \mathbf{Y}$$

  onde $n \ll m, K$
- Os melhores atributos são encontrados automaticamente de forma a minimizar o custo de treinamento

Função Custo

- Predição: $\hat{\mathbf{Y}} = \mathbf{X}\mathbf{W}^T$
- Função custo:

$$J = J(\mathbf{W}, \mathbf{X}) = \frac{1}{2} \sum_{i=1}^{m} \sum_{k=1}^{K} R_{ik} (\mathbf{w}_k^T \mathbf{x}^{(i)} - y_k^{(i)})^2$$

- Se $\mathbf{X}$ está fixa, sabemos encontrar a solução ótima para $\mathbf{W}$
- Da mesma forma podemos fixar $\mathbf{W}$ e encontrar a solução ótima para $\mathbf{X}$
- Mínimos quadrados alternados *(alternating least squares)*:
  - Parte-se de valores iniciais aleatórios para $\mathbf{W}$ e $\mathbf{X}$
  - Otimiza-se: $\mathbf{W} \to \mathbf{X} \to \mathbf{W} \to \mathbf{X} \to \cdots$
  - A cada iteração o custo nunca pode aumentar, portanto deve convergir

Notação Matricial

- Predição: $\hat{\mathbf{y}}^{(i)} = \mathbf{W}\mathbf{x}^{(i)} \quad \Longrightarrow \quad \hat{\mathbf{Y}}^T = \mathbf{W}\mathbf{X}^T$
- Função custo para o item i:

$$\begin{aligned} J(\mathbf{x}^{(i)}) &= \frac{1}{2}\|(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) \odot \mathbf{R}_{i,:}^T\|^2 \\ &= \frac{1}{2}(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T)^T \operatorname{diag}(\mathbf{R}_{i,:})(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) \end{aligned}$$

- Gradiente:

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{x}^{(i)}} &= \mathbf{W}^T \operatorname{diag}(\mathbf{R}_{i,:})(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) \\ \frac{\partial J}{\partial \mathbf{X}^T} &= \mathbf{W}^T(\mathbf{R}^T \odot (\mathbf{W}\mathbf{X}^T - \mathbf{Y}^T)) \end{aligned}$$

- Obs: lembre-se que $\mathbf{X}_{:,0} = \mathbf{1}$ não é variável de otimização

Com Regularização

- Seja $\mathbf{L} = \mathrm{diag}(0, 1, \ldots, 1) \in \mathbb{R}^{(n+1)\times(n+1)}$
- Função custo para o item i:

$$\begin{aligned} J(\mathbf{x}^{(i)}) &= \frac{1}{2}\|(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) \odot \mathbf{R}_{i,:}^T\|^2 + \frac{\lambda}{2}\mathbf{x}^{(i)T}\mathbf{L}\mathbf{x}^{(i)} \\ &= \frac{1}{2}(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T)^T \mathrm{diag}(\mathbf{R}_{i,:})(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) + \frac{\lambda}{2}\mathbf{x}^{(i)T}\mathbf{L}\mathbf{x}^{(i)} \end{aligned}$$

- Gradiente:

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{x}^{(i)}} &= \mathbf{W}^T \mathrm{diag}(\mathbf{R}_{i,:})(\mathbf{W}\mathbf{x}^{(i)} - \mathbf{Y}_{i,:}^T) + \lambda\mathbf{L}\mathbf{x}^{(i)} \\ \frac{\partial J}{\partial \mathbf{X}^T} &= \mathbf{W}^T(\mathbf{R}^T \odot (\mathbf{W}\mathbf{X}^T - \mathbf{Y}^T)) + \lambda\mathbf{L}\mathbf{X}^T \end{aligned}$$

Alternativa: Otimização Conjunta

- Função custo:

$$J(\mathbf{W}, \mathbf{X}) = \frac{1}{2} \sum_{i=1}^{m} \sum_{k=1}^{K} R_{ik} (\mathbf{w}_k^T \mathbf{x}^{(i)} - y_k^{(i)})^2 + \frac{\lambda}{2} \mathbf{w}_k^T \mathbf{L} \mathbf{w}_k + \frac{\lambda}{2} \mathbf{x}^{(i)T} \mathbf{L} \mathbf{x}^{(i)}$$

- Gradiente:

$$\frac{\partial J}{\partial \mathbf{W}^T} = \mathbf{X}^T (\mathbf{R} \odot (\mathbf{X}\mathbf{W}^T - \mathbf{Y})) + \lambda \mathbf{L} \mathbf{W}^T$$

$$\frac{\partial J}{\partial \mathbf{X}^T} = \mathbf{W}^T (\mathbf{R}^T \odot (\mathbf{W}\mathbf{X}^T - \mathbf{Y}^T)) + \lambda \mathbf{L} \mathbf{X}^T$$

Aplicação: Encontrando itens semelhantes

- O modelo efetivamente aprende uma representação em $\mathbb{R}^n$ para os itens
  - Extração de atributos automática
- De posse desta representação, podemos resolver outros problemas como:
  - Encontrar itens semelhantes a um dado item:

$$\min_{j \neq i} \|\mathbf{x}^{(j)} - \mathbf{x}^{(i)}\|$$

  - Encontrar grupos de itens semelhantes (clustering)