matrizes retangulares e SVD

Jacques Wainer

30/3/20

Um outro livro texto:
https://www.ufrgs.br/reamat/AlgebraLinear/livro/main.html

1 Matrizes retangulares

Matrizes não quadradas podem ser pensadas como transformações lineares entre espaços de dimensões diferentes.

Mas para esse curso não há muita vantagem em pensar nesses termos.

1.1 Matriz como dados

Matrizes retangulares são a forma de representar dados para aprendizado de maquina.

Cada linha representa um dado e cada coluna os atributos/coordenadas/features desse dado.

Um vetor dentro da matrix é uma linha (um ponto no espaço c dimensional onde c é o número de colunas.

Isso vai contrario a pratica em algebra linear que vetores são colunas. Umas poucas pessoas e comunidades em aprendizado de maquina trocam a ordem e cada coluna representa um dado.

Normalmente temos mais dados que atributos (medimos 40 coisas sobre cada pessoa e temos 1000 pessoas) Nesse caso a matriz é alta e fina (alta n=1000 e fina c=40).

Para imagens talvez valha a apena trocar linhas por colunas - cada coluna é uma imagem e cada linha os pixeis dessa imagem. Uma imagem de 1000 por 1000 tem 1.000.000 de atributos e voce provavelmente so tem 500 imagens.

Neste caso a matriz ainda é alta (1.000.000) e fina (500).

Em bioinformática, de vez em quando as matrizes sao baixas e largas (muitos atributos - genes para poucos dados - seres)

2 Normalizações dos dados

2.1 Media 0

Um procedimento comum com dados (mas nao com matrizes em geral) é normalizar cada coluna dos dados

- calcule a media de cada coluna
- subtraia a média de cada dado de uma coluna

Assim cada coluna tera média 0

2.2 Normalização

- calcule a média e o desvio padrão de cada coluna

- subtraia a média de cada dado de uma coluna
- divida pelo desvio padrão

Desta forma cada coluna terá media 0 e desvio padrão 1

Há ouros preprocessamentos úteis para aprendizado de maquina (que veremos em outra disciplina)

3 Rank de uma matrix retangular

O rank de uma matrix quadrada é a dimensão do subespaço que é a imagem da transformação e isso e' o numero de colunas linearmente independentes de A

nao vimos isso mas é também o numero de linhas linearmente independentes de A

Para uma matriz retangular, o rank é o numero de linhas ou de colunas que sao linearmente independentes (o menor dos 2)

No caso de matrizes altas e finas é o numero de colunas linearmente independentes

4 Sigular value decomposition SVD

Videos https://www.youtube.com/playlist?list=PLMrJAkhIeNNSVjnsviglFoY2nXildDCcv (1 2 e 3 e possivelmente o de PCA e de truncation - mas com tempo veja todos)

Note que ele não usa o nosso padrão que cada linha é um dado. Para ele cada coluna é um dado mas a matrix é ainda alta e fina.

O equivalente de autovetores e autovalores para matrizes retangulares

$$A = UDV^T$$

A é a matriz dos dados nxc

- n dados
- cada um com c (de colunas) atributos

D é uma matriz diagonal com os "singular values" (autovalores)

U e V são ortogonais (colunas são ortonormais) e $U^{-1} = U^T$ e $V^{-1} = V^T$

U e V sao as matrizes que seria equivalente a "autovetores" mas tem dimensões diferentes

- V^T é c x c

as outras 2 matrizes tem tamanhos diferentes em 2 formulações diferentes.

4.1 full matrix

Na primeira formulação (do wikipedia por exemplo), que nós chamamos de *full matrix*

- U é n x n

- D é n x c

- V^T é c x c

https://intoli.com/blog/pca-and-svd/

4.2 Formulação compacta

- U é n x c

- D é c x c
- V^T é c x c

https://public.lanl.gov/mewall/kluwer2002.html

As linhas a mais da matriz D são todas 0. E portanto as colunas a mais da matriz U não contribuem para os valores da matriz A

4.3 D

A matriz D é uma matrix diagonal onde todos os singular values são positivos (ou 0) e sao ordenados em ordem decrescente

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_c \geq 0$$

4.4 U e V

U sao os autovetores da matriz AA^T

V sao os autovetores da matriz $A^T A$

tanto $A^T A$ como AA^T são simétricas e portanto seus autovetores são ortonormais.

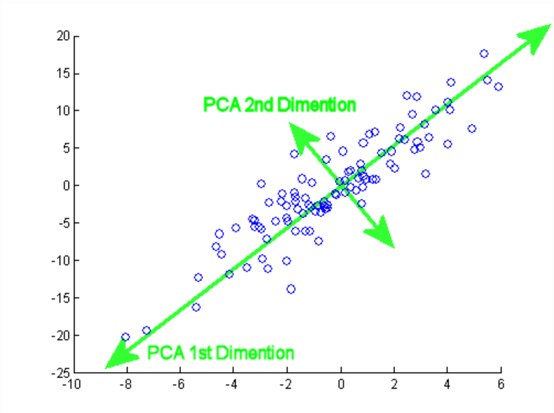
4.5 Algoritmos

Ha 3 grandes algoritmos apra calcular o SVD

- decomposição QR da matrix A
- autovetores e autovalores da matrix de covariância de A ($A^T A$ - se os dados estiverem com media 0)
- random projections (random SVD)

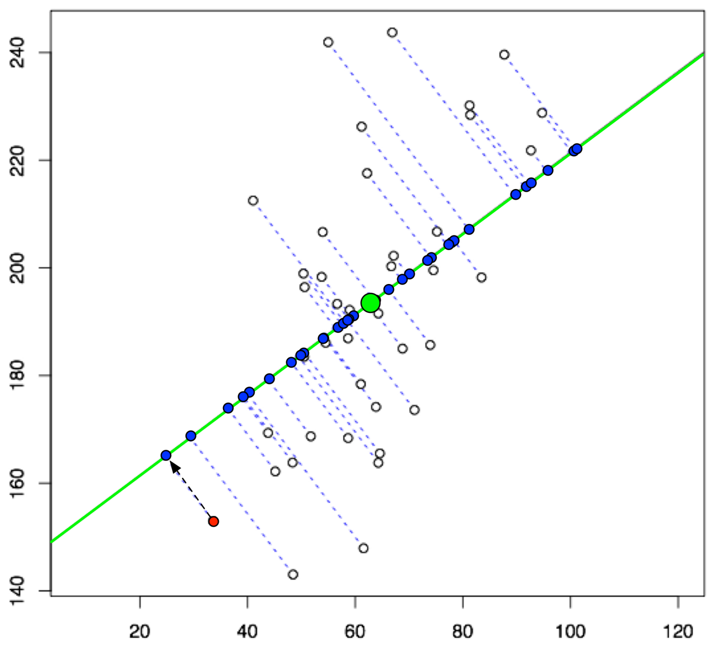
5 Redução de dimensionalidade de A (PCA)

Reduzir o número de atributos para cada dado



Manter as direções/subespaços com maior variabilidade

Projetar os dados nesses subespaços



5.1 SVD truncado - low rank decomposition

manter $r < c$ dimensões dos dados.

usar apenas - as primeiras r colunas de U e - a submatrix rxr de D - as primeiras r linhas de V^T

$$A \approx A_r = U_r D_{rxr} V_r^T$$

- U_r tem dimensão nxr
- D_{rxr} rem dimensão rxr
- V_r^T tem dimensão rxc
- portanto A_r tem dimensão nxc as mesmas dimensões que A mas com rank r.

A_r é a melhor aproximação de rank r da matriz A

5.2 Dados projetados e dados reconstruidos e eigenfaces/eigendados

para a conveção que dados são linhas

A_r são os dados **reconstruídos**. Ou seja são os dados no mesmo formato que os de A

$U_r D_{rxr}$ são os dados **projetados** no subespaço Sao dados com apenas r atributos

Os dados projetados é que voce usa em aplicações que se seguem, por exemplo, de aprendizado de maquina.

As linhas da matriz V_r^T são as bases do subespaço onde estamos projetando os dados. Eles representam (na ordem) as padrões básicos dos dados

Note que na serie de videos indicada, eles usam a convenção que dados dao colunas, o que muda quem sao os dados projetados e quem sao dos eigendados.

5.3 Como escolher a truncagem (r)

- voce sabe de antemão. Em texto usa-se normalmente 50 ou 100 dimensões

- escolha as linhas cujos singular values > 1

- o singular value é também a proporção da variância total dos dados capturados por cada autovetor (linha) de `V^T_`. Assim selecione os singular values que somam x% da soma dos singular values. Normalmente usa-se 80% (corajoso) ou 95% (menos corajoso).

5.4 Dados com media 0

O SVD truncado encontra o **subespaço** de dimensão r que melhor aproxima os dados. Para que isso represente as direçoes que contem as maiores variações dos dados, é preciso que a media dos dados seja 0.

6 Aspectos computacionais

6.1 Complexidade

https://en.wikipedia.org/wiki/Computational_complexity_of_mathematical_operations

Multiplicação de matrizes quadradas $O(n^3)$ mas teoricamente $O(n^2.4)$

-Inversão de matrizes = multiplicação

-SVD (matrix nxc = $O(nc^2)$ compacto, $O(nc^2 + cn^2)$ full matrix

6.2 Matrizes esparsas

Comum matrizes com muitos 0.

Representar essas matrizes da forma tradicional pode ser pouco eficiente.

Há varias formas de representar essas matrizes
https://en.wikipedia.org/wiki/Sparse_matrix

6.3 Bibliotecas de algebra linear

Implementa algoritmos básicos de algebra linear, como produto de matrix pro vetor, produto escalar, produto de matrizes, autovalores, fatorizações

- Otimizado para diferentes condições das matrizes (simétrica, esparsa, definida positiva - autovalores positivos, etc).

- Otimizado para diferentes tamanhos

- Otimizado para diferentes arquiteturas (entende de paralelismo na CPU, entende de comandos de linguagem de maquina, entende de cache L1 e L2)

Históricas

- LAPACK
- IMSL
- BLAS

Modernas

- OpenBLAS
- ATLAS
- MKL
- Armadilo
- Eigen

https://en.wikipedia.org/wiki/Comparison_of_linear_algebra_libraries

7 Numpy e scipy

Matrizes nao são primitivas em python (mas são em R, matlab, Julia)

Numpy implementa matrizes de forma eficiente. Pacote externo a implementação standard do python

Numpy implementa funções em matrizes eficientemente. Escrever **for** para manipular matrizes deve ser evitado (muito lento)

Funções de numpy evitam as restrições de multithread do python

numpy utiliza as bibliotecas básicas do algebra linear (MKL ou ATLAS)

manual numpy https://numpy.org/doc/stable/reference/routines.html

7.1 Scipy

pacote externo do python que implementa funções cientificas -

scipy implementa funções de otimização que veremos adiante

reimplementa algumas funções de algebra linear

implementa algumas funçoes

7.2 numpy

- transposta
- leitura
-

7.3 numpy.linalg

https://numpy.org/doc/stable/reference/routines.linalg.html

- svd
- autovalores (eigen)
- outras fatorações
- produto de matrizes
- norma (tamanho)
- inversa de matrizes
- determinantes
- resolução de equações lineares

7.4 scipy

doc https://docs.scipy.org/doc/scipy/reference/

matrizes esparsas https://docs.scipy.org/doc/scipy/reference/sparse.html NAO TEM em numpy

algebra linear para matriz esparsas https://docs.scipy.org/doc/scipy/reference/sparse.linalg.html

algebra linear nao esparsa https://docs.scipy.org/doc/scipy/reference/linalg.html

7.5 SVD truncado e PCA

usar o sklearn - pacote externo do python para aprendizado de maquina.

truncated SVD https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html mais basico, voce precisa dar o r **n_components**

PCA https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

7.6 Numba e Cython

cython (C em sintaxe de python que pode ser incorporado em python)
https://cython.org/

video numba https://www.youtube.com/watch?v=x58W9A2lnQc