

Mínimos Quadrados e Análise de Componentes Principais

Prof. Dr. José Alberto Cuminato
Disciplina: Fundamentos de Matemática Aplicada

Alunos:

Cristian Júlio de Barros	/	NºUSP	3107731
João Carlos Batista	/	NºUSP	6792197

São Carlos

Jan/2018

Resumo

No estudo de grandes bases de dados são consideradas técnicas para redução da dimensionalidade dos dados com o propósito de se realizar a análise desses dados multivariados. Para este trabalho, utilizaremos duas técnicas: método dos mínimos quadrados para ajustar um hiperplano que melhor aproxime os dados, projetando-os sobre este hiperplano é também cálculo das componentes principais, projetando os dados no subespaço coberto pelas duas componentes principais. Técnicas de decomposição de matrizes como a decomposição de Francis (QR), decomposição em valores singulares e também a decomposição espectral serão consideradas.

Palavras-chave: Método dos quadrados mínimos, decomposição QR, decomposição SVD, análise de componentes principais (PCA)

Sumário

1. INTRODUÇÃO	4
2. JUSTIFICATIVA	4
3. OBJETIVO.....	4
4. MATERIAIS E MÉTODOS DE ANÁLISE.....	4
4.1 SVD alternativo	5
5. Banco de Dados Utilizados na Implementação dos Algoritmos:	6
5.1 Base: auto – mpg.data	6
5.2 Base: household power consumption	6
6 Mínimo Quadrados Resultados.....	7
6.1 Base: auto mpg data	7
6.2 Base: Household power consumption	10
7 PCA Resultados	14
7.1 Base: auto mpg data	14
7.2 Base: household power consumption	18
8 Conclusões.....	22
9 Referências.....	23
10 Anexos – Algoritmo em octave	23
10.1 Autovetores	23
10.2 SVD	23
10.3 QR.....	23
10.4 plot line direction.....	24
10.5 gerar gráfico mq	24
10.6 gerar gráfico pca	27
10.7 PCA	30
10.8 MQ	31
10.9 Norma ou desvio padrão	32
10.10 Normalizar dados	32
10.11 Carregar base.....	33
10.12 Comparar SVD Octave X SVD Alternativo	33
11 Chamadas de funções implementada no MQ	35
12 Chamadas de funções implementada no PCA	36

1. INTRODUÇÃO

Para aplicação dos temas mínimos quadrados, autovalores e autovetores, decomposição SVD e análise de componentes principais (PCA) foram escolhidos os bancos de dados disponíveis no <http://archive.ics.uci.edu/ml/>:

- auto-mpg;
- household electric power consumption;

PRINCIPAL COMPONENT ANALYSIS (PCA): A ideia é encontrar uma base para representar os dados, de modo a minimizar a redundância de informação na representação.

MÍNIMOS QUADRADOS (MQ): A ideia consiste em procurar uma aproximação f polinomial de grau m ($m \ll n$) que minimiza o erro médio quadrático $\sum_{i=0}^n [y_i - f(x_i)]^2$. O mesmo critério de minimização poderá ser aplicado a classes de funções não polinomiais.

SINGULAR VALUE DECOMPOSICAO (SVD): A ideia é transformar uma matriz não quadrada m x n em uma matriz quadrada m x m.

2. JUSTIFICATIVA

Fazendo uso dos temas mencionados na introdução podemos analisar o comportamento de um banco de dado quanto as suas categorias e disposição dos dados.

3. OBJETIVO

Aplicar os bancos de dados escolhidos, para análise dos temas propostos quanto a sensibilidade e desempenho das técnicas. Esse não terá o objetivo a análise e resultado dos dados, somente aplicação das técnicas e análise das mesmas em condições adversas como tipos de dados.

4. MATERIAIS E MÉTODOS DE ANÁLISE

Foram implementados os algoritmos de SVD, Mínimos quadrados e PCA na ferramenta Octave conforme scripts disponibilizados nos anexos.

4.1 SVD alternativo

A rotina de cálculo da decomposição SVD desenvolvida em alternativa a função original existente no Octave, permite a parametrização do valor de tolerância ao erro e a quantidade máxima de iterações que podem ser executadas no algoritmo QR iterativo (método de Francis).

Para validar o algoritmo alternativo foram criadas matrizes de diferentes tamanhos e para cada uma delas o SVD alternativo foi executado com diferentes valores para o parâmetro que define a tolerância. Como medida de conformidade do SVD alternativo foi efetuada uma soma de todos os valores absolutos resultantes da subtração da matriz de autovalores fornecida pelo Octave menos a matriz de autovalores do SVD alternativo.

Tabelas de diferenças entre SVD do Octave X SVD Alternativo							
Tamanho da matriz nxn	Valor de tolerancia no calculo do SVD Alternativo						
	1,00E-03	1,00E-04	1,00E-05	1,00E-06	1,00E-07	1,00E-08	1,00E-09
1	0	0	0	0	0	0	0
2	1,24842E-07	5,8752E-09	1,30E-11	6,13E-13	1,44E-15	2,22E-16	1,11E-16
3	2,12463E-07	2,04581E-10	6,36E-12	1,98E-13	1,47E-15	1,69E-15	1,69E-15
4	1,39346E-05	2,61126E-07	1,72E-09	1,13E-11	2,04E-13	1,94E-15	1,61E-15
5	1,57343E-06	9,98499E-09	6,34E-11	2,17E-12	1,64E-14	4,62E-15	4,62E-15
6	2,15393E-06	2,03575E-08	1,92E-10	1,83E-12	2,57E-14	8,58E-15	8,58E-15
7	1,2816E-06	3,31224E-08	1,97E-10	2,42E-12	3,19E-14	6,34E-15	6,22E-15
8	1,88208E-05	1,94765E-07	2,01E-09	1,64E-11	1,73E-13	5,15E-15	6,04E-15
9	1,78288E-06	1,77651E-08	1,77E-10	1,77E-12	2,63E-14	1,16E-14	1,16E-14
10	1,50912E-06	1,5081E-08	1,43E-10	1,48E-12	2,48E-14	1,00E-14	1,00E-14
11	8,28056E-06	7,75502E-08	8,53E-10	8,02E-12	1,02E-13	2,89E-14	2,87E-14
12	4,51023E-06	3,61217E-08	3,70E-10	2,84E-12	4,10E-14	1,53E-14	1,53E-14
13	6,05549E-06	5,29688E-08	5,57E-10	5,87E-12	7,13E-14	1,00E-14	1,00E-14
14	1,67149E-05	1,71923E-07	1,64E-09	1,68E-11	1,99E-13	3,02E-14	2,97E-14
15	3,77632E-06	3,78156E-08	3,54E-10	3,63E-12	5,27E-14	2,01E-14	2,01E-14
16	2,29067E-06	3,08659E-08	6,04E-10	5,92E-12	1,21E-13	6,71E-14	6,71E-14
17	1,17949E-05	1,17871E-07	1,18E-09	1,18E-11	1,38E-13	2,37E-14	2,37E-14
18	3,46501E-06	3,62282E-08	3,47E-10	3,64E-12	5,60E-14	2,24E-14	2,24E-14
19	0,000184416	4,24685E-07	4,25E-09	4,26E-11	4,55E-13	4,35E-14	4,35E-14
20	4,6096E-05	6,42122E-07	6,03E-09	6,29E-11	6,12E-13	2,76E-14	2,52E-14
21	3,3023E-06	3,33333E-08	3,18E-10	3,35E-12	7,41E-14	4,39E-14	4,39E-14
22	2,36985E-05	5,64232E-07	5,45E-09	5,43E-11	5,72E-13	3,63E-14	3,14E-14
23	8,18714E-06	7,49877E-08	7,52E-10	7,62E-12	1,48E-13	7,63E-14	7,63E-14
24	1,46869E-06	1,42624E-08	1,54E-10	1,52E-12	4,25E-14	4,48E-14	4,48E-14
25	1,06761E-05	1,05932E-07	1,05E-09	1,08E-11	4,45E-13	3,41E-13	3,41E-13
26	1,99829E-06	1,89289E-08	1,99E-10	1,94E-12	7,72E-14	7,54E-14	7,54E-14
27	4,26903E-06	2,84167E-08	2,47E-10	2,53E-12	1,62E-13	1,47E-13	1,47E-13
28	6,74598E-06	6,73545E-08	6,72E-10	7,08E-12	1,39E-13	7,44E-14	7,44E-14
29	8,31354E-06	9,61528E-08	1,64E-09	3,06E-11	4,39E-13	8,18E-14	8,07E-14
30	1,19947E-05	1,11464E-07	1,10E-09	1,12E-11	1,79E-13	8,16E-14	8,16E-14

Os valores da tabela acima mostram que o SVD alternativo possui resultados semelhantes ao obtidos com o SVD original do Octave, para a execução do SVD para os bancos de dados selecionados foi utilizado um valor de tolerância igual à 1,00E-12.

5. Banco de Dados Utilizados na Implementação dos Algoritmos:

5.1 Base: auto – mpg.data

Os dados dizem respeito ao consumo de combustível em milhas por galão, a ser prevista em termos de 3 multivalores discreto e 5 atributos contínuos. (Quinlan, 1993).

Este conjunto de dados é uma versão ligeiramente modificada do conjunto de dados fornecido na biblioteca StatLib. Alinhado com o uso por Ross Quinlan (1993) em prever o "mpg" atributo. Foram removidos do original 8 instâncias porque eles tinham valores desconhecidos para o atributo "mpg".

Informações de Atributo:

1. Mpg: continua
2. Cilindros: valores múltiplos discretos
3. Deslocamento: continua
4. Potência em cavalos: continua
5. Peso: continua
6. Aceleração: contínua
7. Ano/modelo: valores múltiplos discretos
8. Origem: valores múltiplos discretos
9. ~~Nome do carro: nominal~~ (retirado da análise)

	A	B	C	D	E	F	G	H	
1	Mpg	Cilindros	Deslocamento	tência em cavalos	Peso	Aceleração	Ano/modelo	Origem	Nome do carro
2	18.0	8	307.0	130.0	3504.	12.0	70	1	chevrolet chevelle malibu
3	15.0	8	350.0	165.0	3693.	11.5	70	1	buick skylark 320
4	18.0	8	318.0	150.0	3436.	11.0	70	1	plymouth satellite
5	16.0	8	304.0	150.0	3433.	12.0	70	1	amc rebel sst
6	17.0	8	302.0	140.0	3449.	10.5	70	1	ford torino
7	15.0	8	429.0	198.0	4341.	10.0	70	1	ford galaxie 500

•
•
•

Base original com 386 instâncias e 9 atributos

Base atual está com 386 instâncias e 8 atributos

5.2 Base: household power consumption

Os dados apresentam a energia ativa consumida em cada minuto (em watts hora) no agregado familiar por equipamentos elétricos.

Informações de Atributo:

1. ~~Data: Data no formato dd/mm/aaaa~~ (retirado da análise)
2. ~~Tempo: tempo no formato hh:mm:ss~~ (retirado da análise)

3. global_active_power: agregado familiar mundial minuto a média potência ativa (em quilowatts)
4. global_reactive_power: agregado familiar mundial minuto a média de potência reativa (em quilowatts)
5. voltage: minutos em média tensão (em volts)
6. global_intensity: agregado familiar intensidade média global de minutos atual (em amperes)
7. sub_metering_1: energia sub-medição no. 1 (em watt-hora de energia ativa). Corresponde à cozinha, que contém principalmente uma máquina de lavar louça, um forno e um micro-ondas (pratos quentes não são elétricos, mas movidos a gás).
8. sub_metering_2: energia sub-medição no. 2 (em watt-hora de energia ativa). Corresponde à lavanderia, contendo uma máquina de lavar roupa, uma queda de cabelo, um frigorífico e uma luz.
9. sub_metering_3: energia sub-medição no. 3 (em watt-hora de energia ativa). Corresponde a um aquecedor elétrico de água e um ar-condicionado.

	X	X	C	D	E	F	G	H	I
1	Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2	16/12/2006	17:24:00	4.216	0.418	234.840	18.400	0.000	1.000	17.000
3	16/12/2006	17:25:00	5.360	0.436	233.630	23.000	0.000	1.000	16.000
4	16/12/2006	17:26:00	5.374	0.498	233.290	23.000	0.000	2.000	17.000
5	16/12/2006	17:27:00	5.388	0.502	233.740	23.000	0.000	1.000	17.000
6	16/12/2006	17:28:00	3.666	0.528	235.680	15.800	0.000	1.000	17.000
7	16/12/2006	17:29:00	3.520	0.522	235.020	15.000	0.000	2.000	17.000

Base original com 2075261 instâncias e 9 atributos

Base atual está com 2000 instâncias e 7 atributos (melhorar o processamento)

6 Mínimo Quadrados Resultados

6.1 Base: auto mpg data

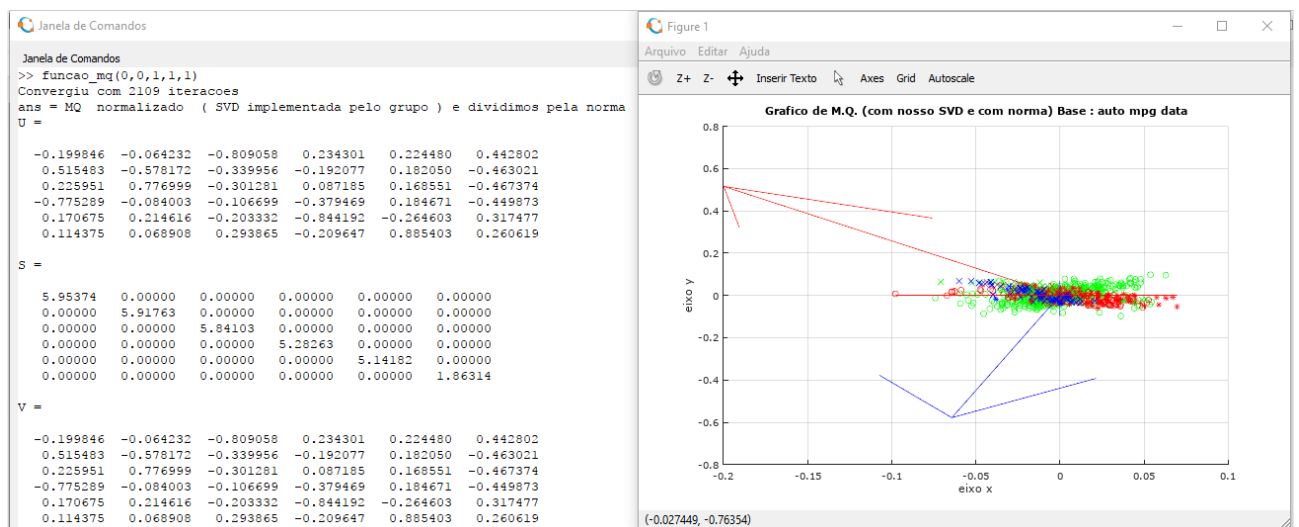


Figura 1: MQ com nosso SVD e utilização da norma.

A figura 1 mostra que os dados estão mais juntos e próximos ao valor zero. O vetor principal, da qual representa a tendência dos dados está representado pela cor vermelha e com valores negativos para ambos os vetores.

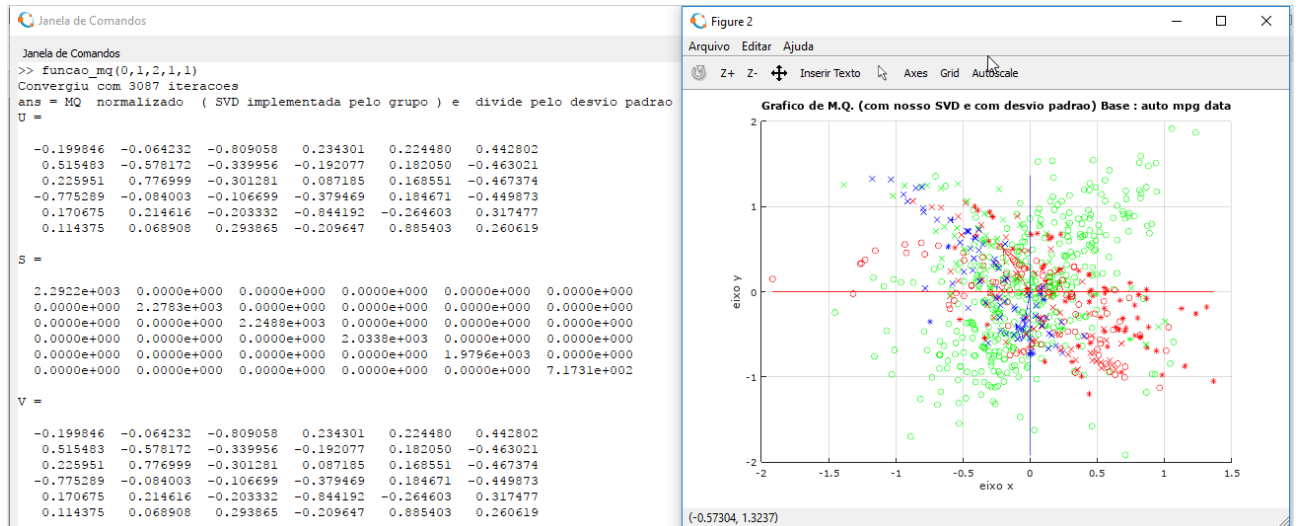


Figura 2: MQ com nosso SVD e utilização do desvio padrão.

A Figura 2 mostra que os dados estão mais espalhados impedindo a melhor visualização da tendência dos dados. O vetor vermelho indica uma certa tendência para o segundo quadrante. Os vetores apresentam valores negativos.

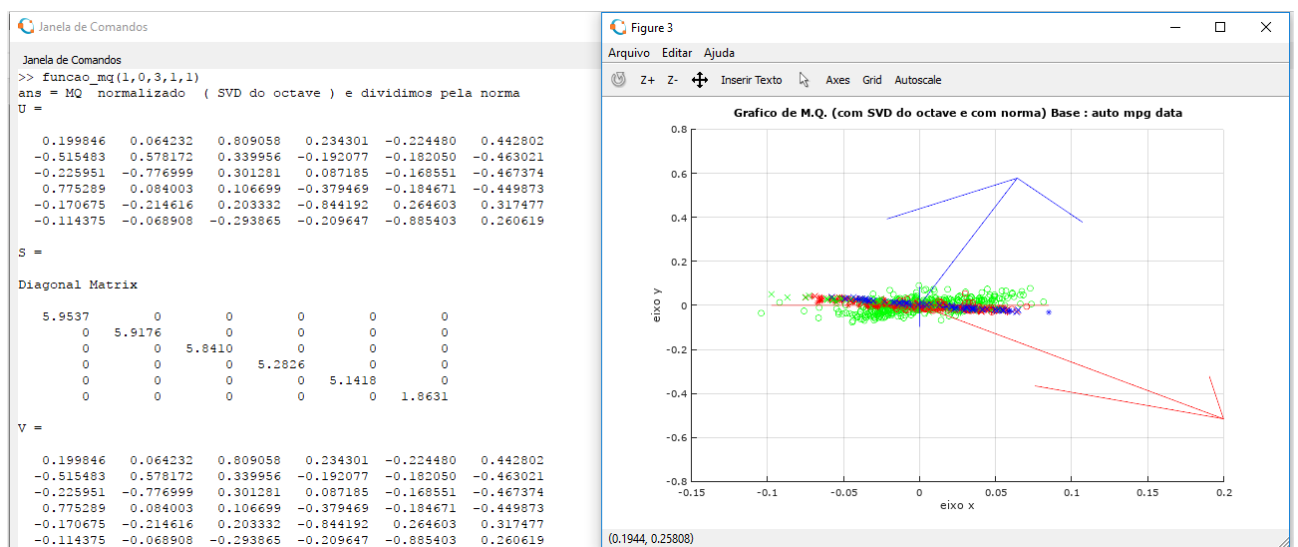


Figura 3: MQ com SVD do Octave e utilização da norma.

A figura 3 mostra que os dados estão mais juntos e próximos ao valor zero. O vetor principal, na qual representa a tendência dos dados está representada pela cor vermelha e apresenta valores positivos para ambos os vetores.

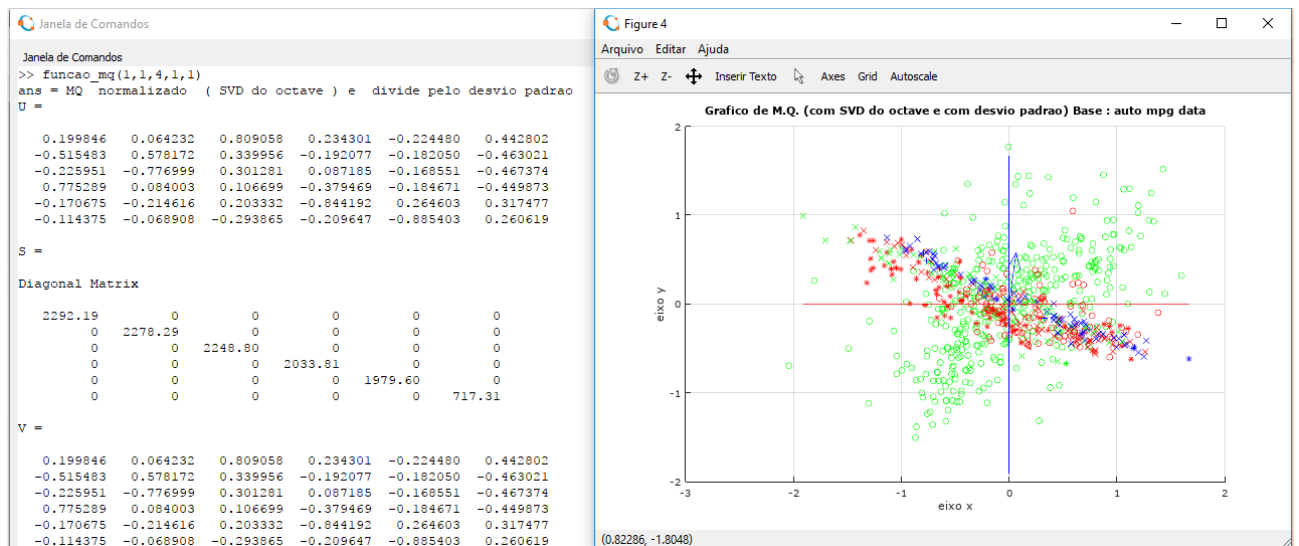


Figura 4: MQ com SVD do Octave e utilização do desvio padrão.

A Figura 4 mostra que os dados estão mais espalhados impedindo a melhor visualização das tendências dos dados. O vetor vermelho indica uma certa tendência para o quarto quadrante. Os vetores apresentam valores diferentes, porém o principal possui um $x > 0$ e $y < 0$.

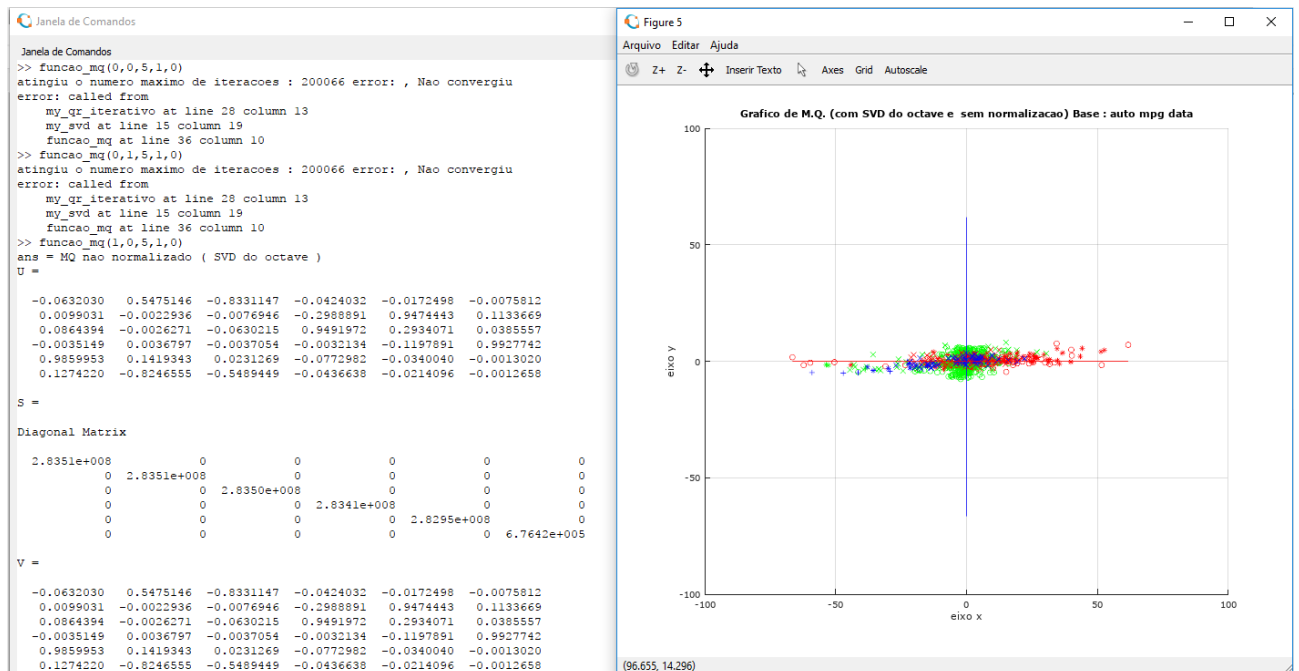


Figura 5: MQ com SVD do Octave e sem norma.

A Figura 5 mostra que os dados estão muito condensados e agrupados próximos a zero. Não se pode ver os vetores principais para analisar a tendência dos dados.

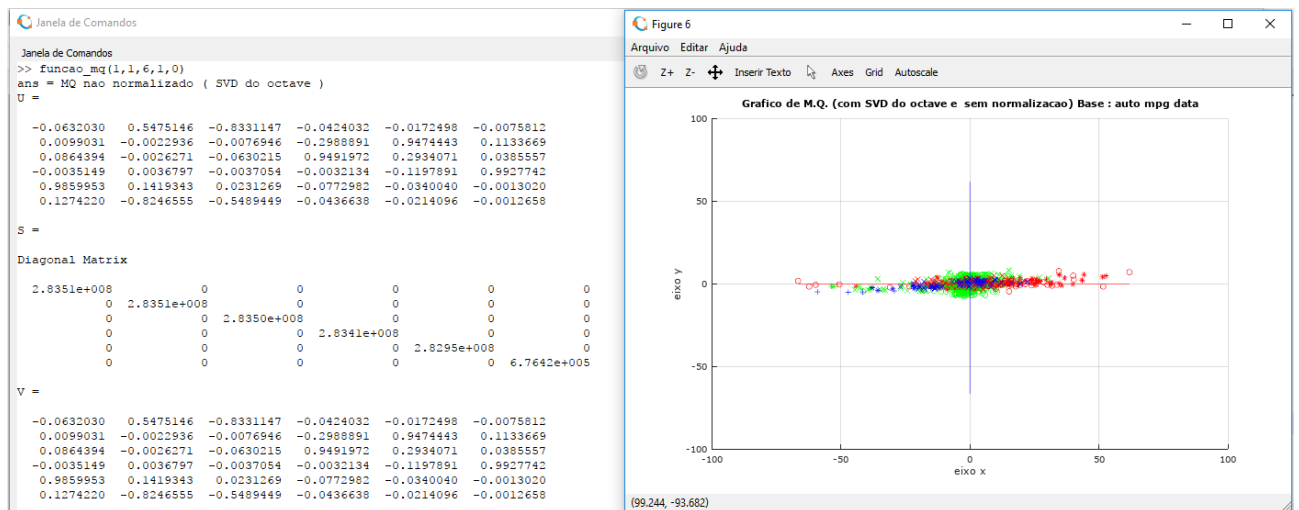


Figura 6: MQ com SVD do Octave e sem norma.

A Figura 6 é semelhante à figura 5, pois mostra que os dados estão muito condensados e agrupados próximos a zero. Não se pode ver os vetores principais para analisar a tendência dos dados.

6.2 Base: Household power consumption

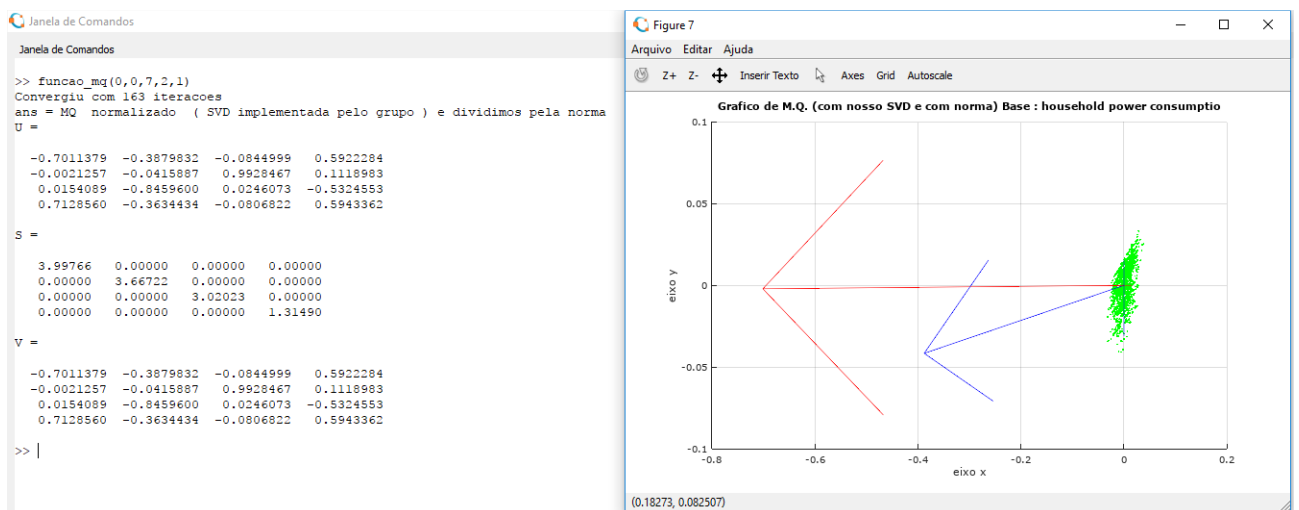


Figura 7: MQ com nosso SVD e utilização da norma para a base Household power consumption.

A Figura 7 mostra que os dados estão muito próximos a zero e condensados, porém observa-se uma certa Tendência, observada pelo vetor vermelho, de seguir exatamente o eixo X.

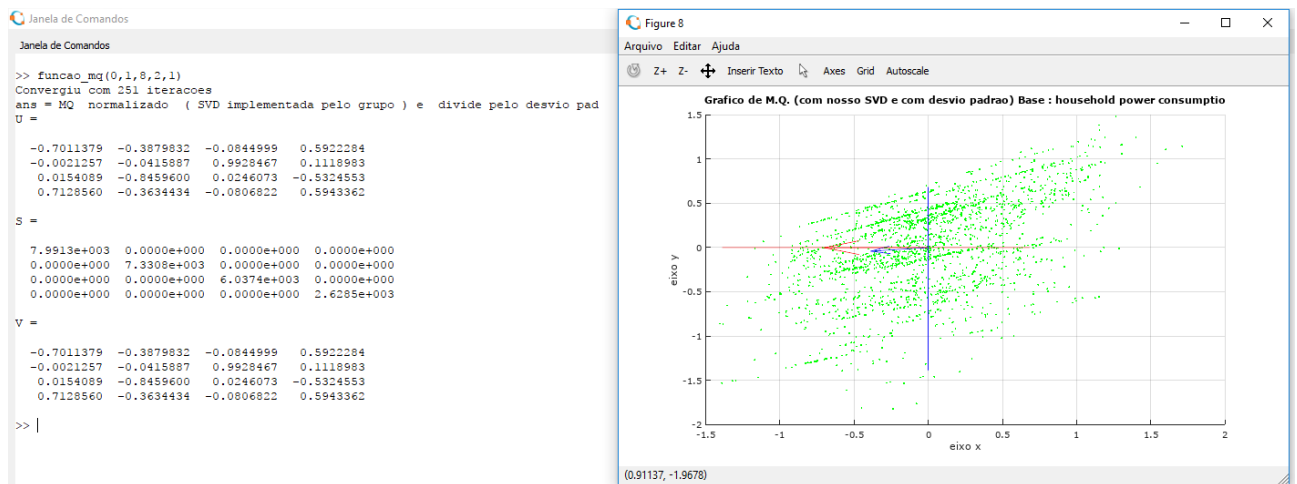


Figura 8: MQ com nosso SVD e utilização do desvio padrão para a base Household power consumption.

A Figura 8 nos mostra que os dados estão bem espalhados, porém é possível observar uma tendência para o eixo X, observada pelo vetor vermelho.

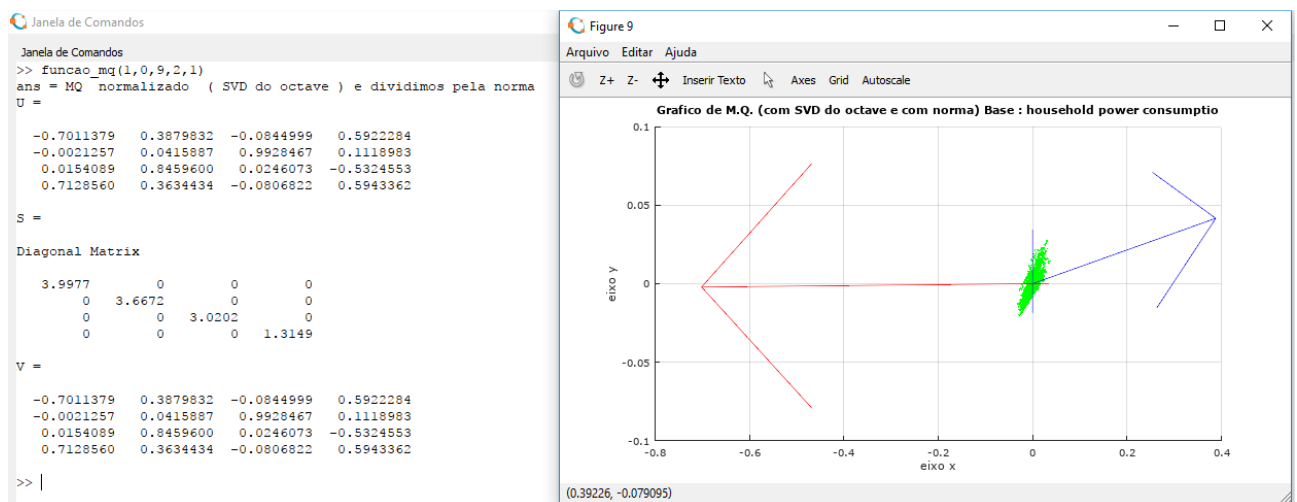


Figura 9: MQ com SVD do Octave e utilização da norma para a base Household power consumption.

A Figura 9 nos mostra que os dados estão muito próximos ao centro com leve tendência para os quadrantes 2 e 3. O vetor vermelho segue o eixo X para o lado esquerdo, entre os quadrantes 2 e 3.

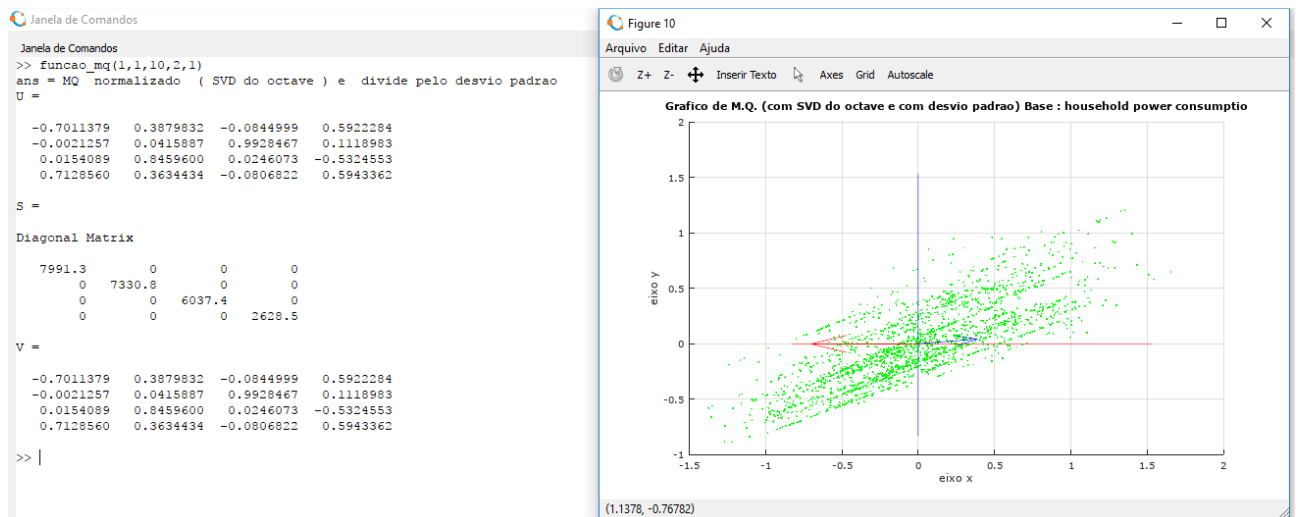


Figura 10: MQ com SVD do Octave e utilização do desvio padrão para a base Household power consumption.

A Figura 10 apresenta os dados bem espalhados variando no eixo X de -1 a 1 e no eixo Y de -0.5 a 0.5. O vetor principal segue o eixo X no sentido negativo, entre os quadrantes 2 e 3.

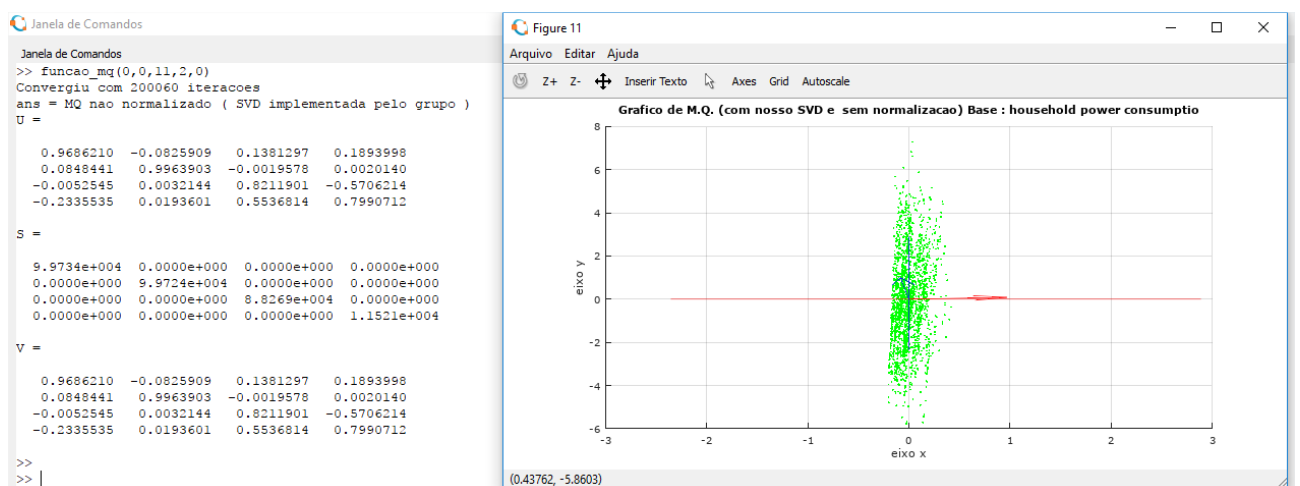


Figura 11: MQ com nosso SVD e sem normalização para a base Household power consumption.

A Figura 11 mostra que os dados estão organizados de forma vertical, ou seja, ao longo do eixo y, porém o vetor principal aponta para o eixo x, provavelmente pela quantidade e peso dos dados ligados a ele.

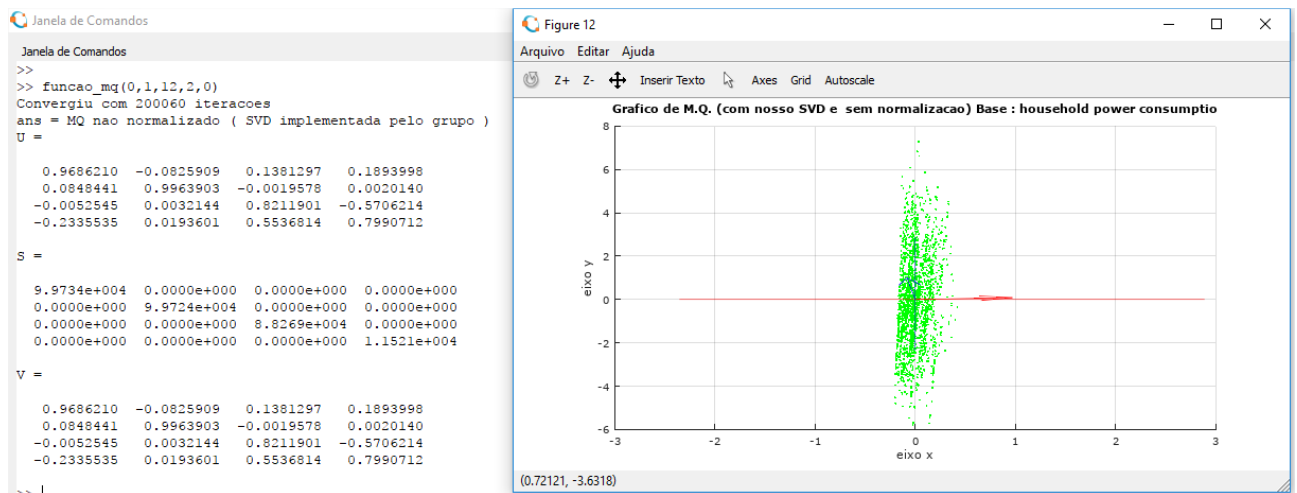


Figura 12: MQ com nosso SVD e sem normalização para a base Household power consumption.

Muito semelhante ao gráfico da Figura 11, pelos mesmos motivos.

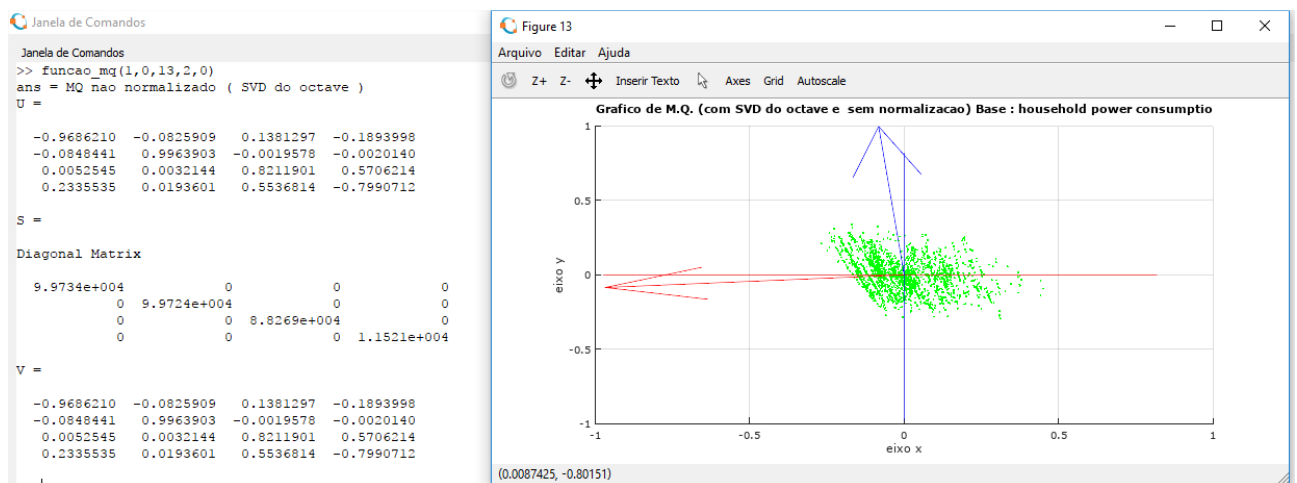


Figura 13: MQ com SVD do Octave e sem utilização da norma para a base Household power consumption.

A Figura 13 nos mostra que os dados estão condensados próximos à coordenada (0,0), porém a tendência dos dados é para o eixo X negativo.

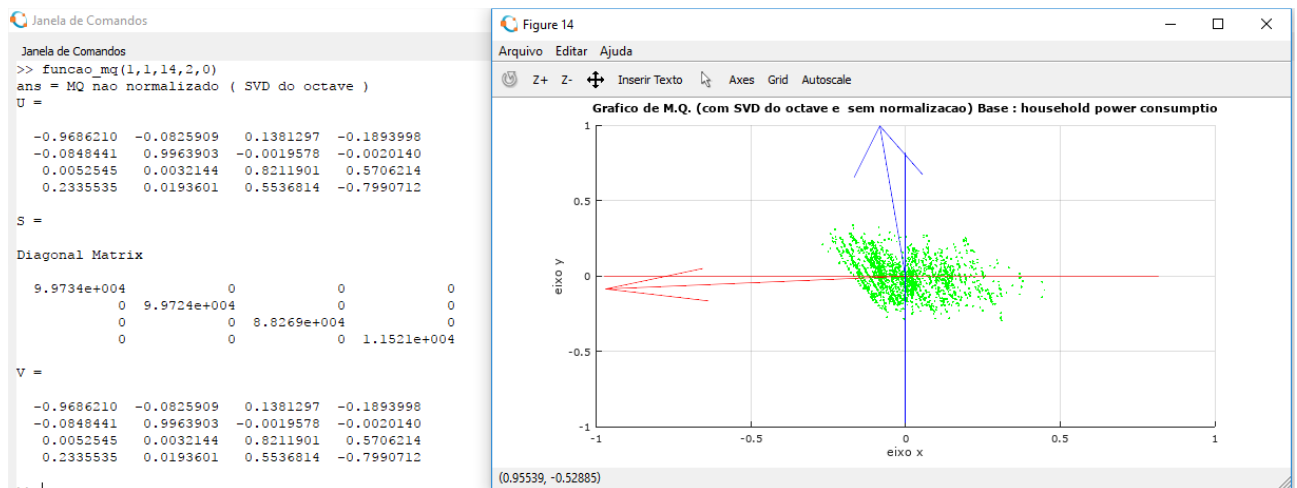


Figura 14: MQ com SVD do Octave e sem utilização da norma para a base Household power consumption.

Semelhante à Figura 13, a Figura 14 apresenta os mesmos resultados.

7 PCA Resultados

7.1 Base: auto mpg data

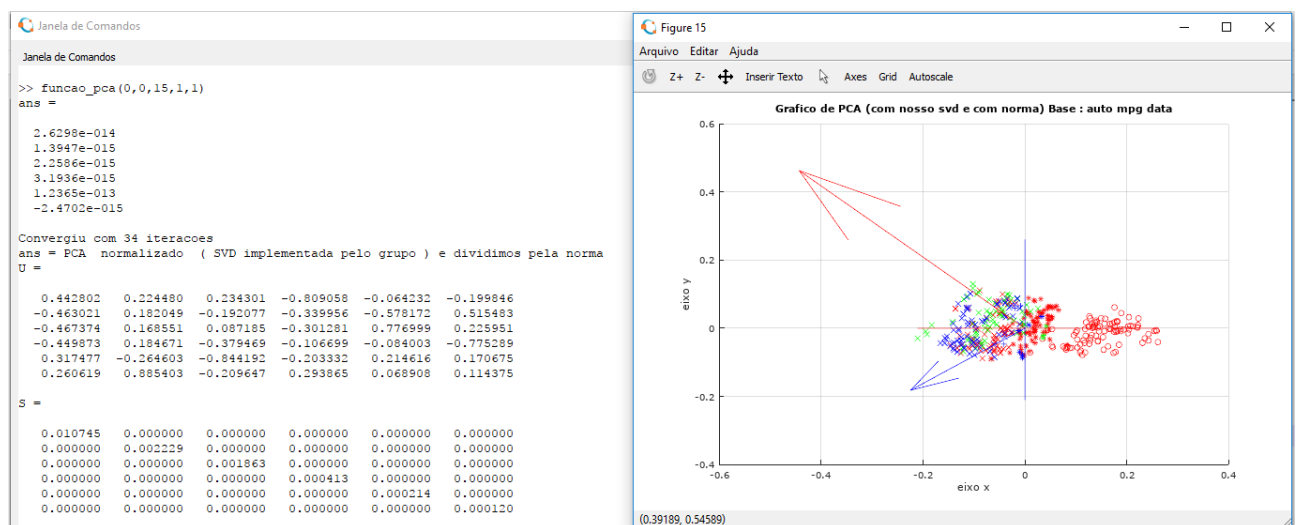


Figura 15: PCA com nosso SVD e utilização da norma para a base auto mpg data.

A Figura 15 mostra que os dados estão levemente espalhados na região da origem das coordenadas. A tendência dos dados está voltada para o segundo quadrante.

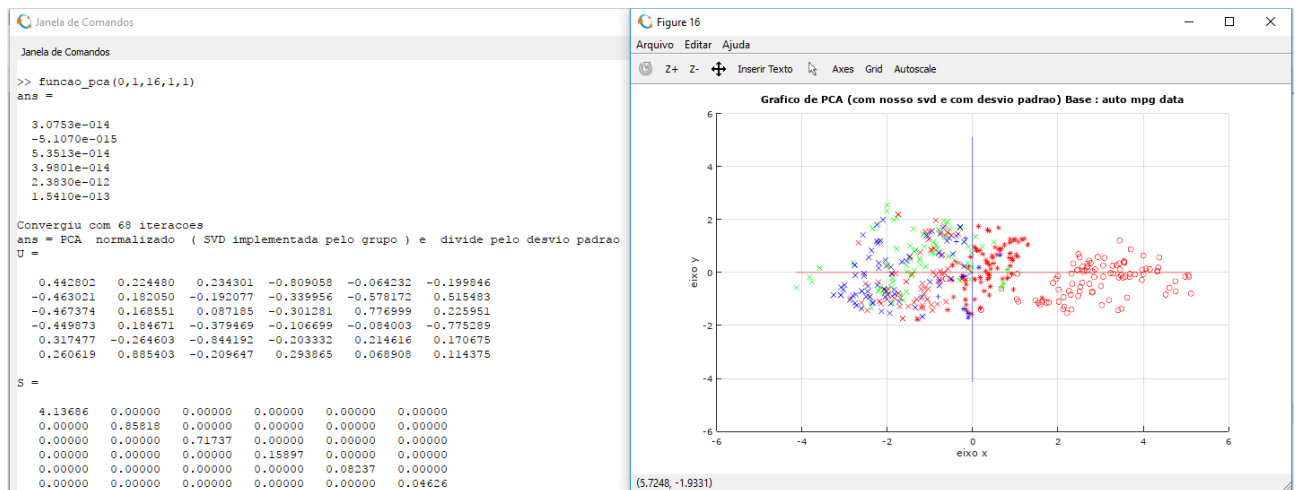


Figura 16: PCA com nosso SVD e utilização do desvio padrão para a base auto mpg data.

A Figura 16 mostra que os dados estão bem espalhados e o vetor de tendência quase não se consegue definir. Com um olhar mais apurado pode-se ver que o vetor de tendência está no segundo quadrante.

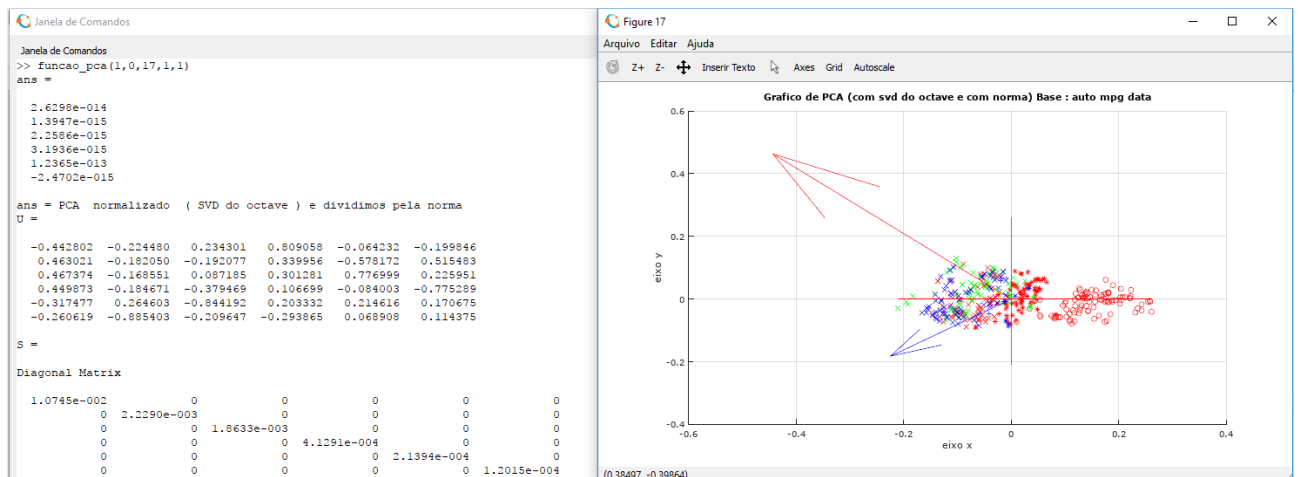


Figura 17: PCA com SVD do Octave e utilização da norma para a base auto mpg data.

A Figura 17 mostra que os dados estão levemente espalhados na origem dos eixos e o vetor de tendência está apontando para o segundo quadrante, demonstrando maior relevância dos dados nessa área.

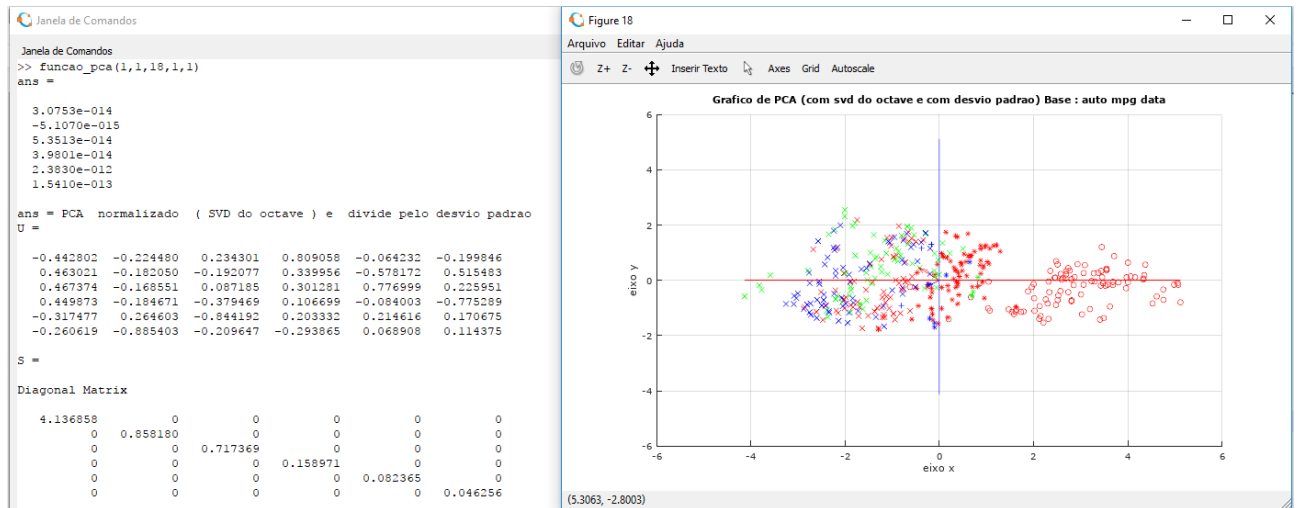


Figura 18: PCA com SVD do Octave e utilização do desvio padrão para a base auto mpg data.

A Figura 18 mostra que os dados estão bem espalhados e por isso que o vetor de tendências não pode ser definido. Com uma visão mais apurada pode-se ver que o vetor está apontando para o segundo quadrante.

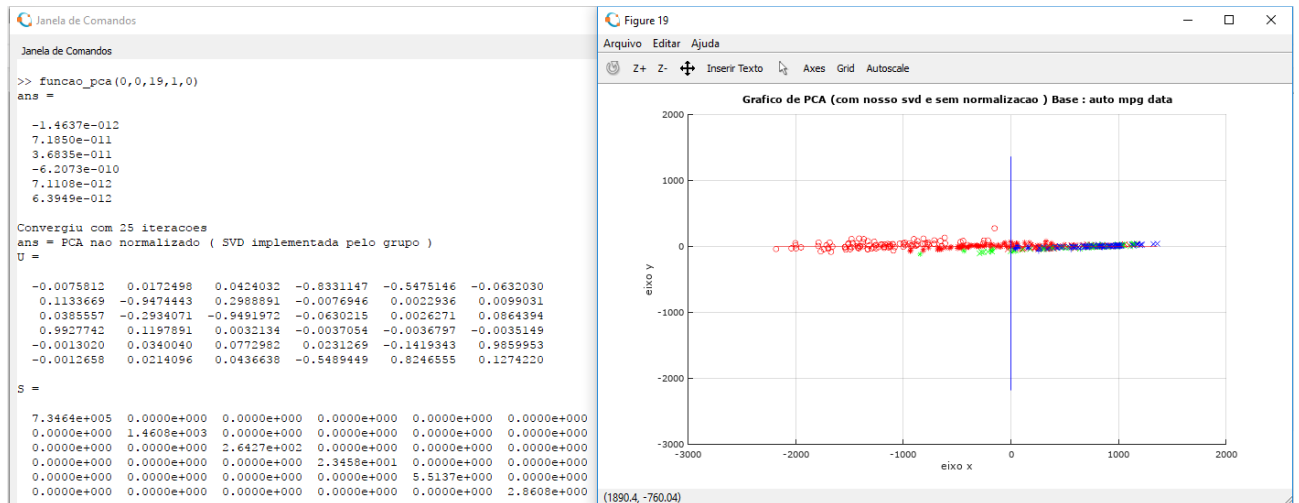


Figura 19: PCA com nosso SVD e sem utilização da norma para a base auto mpg data.

A Figura 19 mostra que os dados estão muito centralizados no eixo X e por conta dessa compactação dos dados não se pode determinar o vetor de tendências.

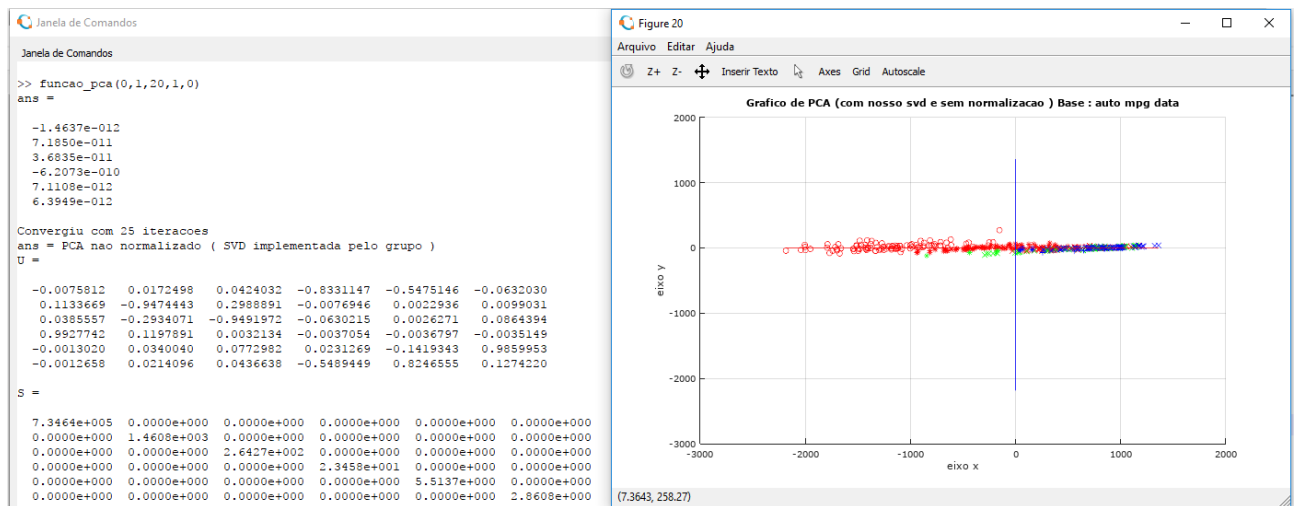


Figura 20: PCA com nosso SVD e sem normalização para a base auto mpg data.

A Figura 20 é semelhante à Figura 19 com indefinição do vetor de tendências.

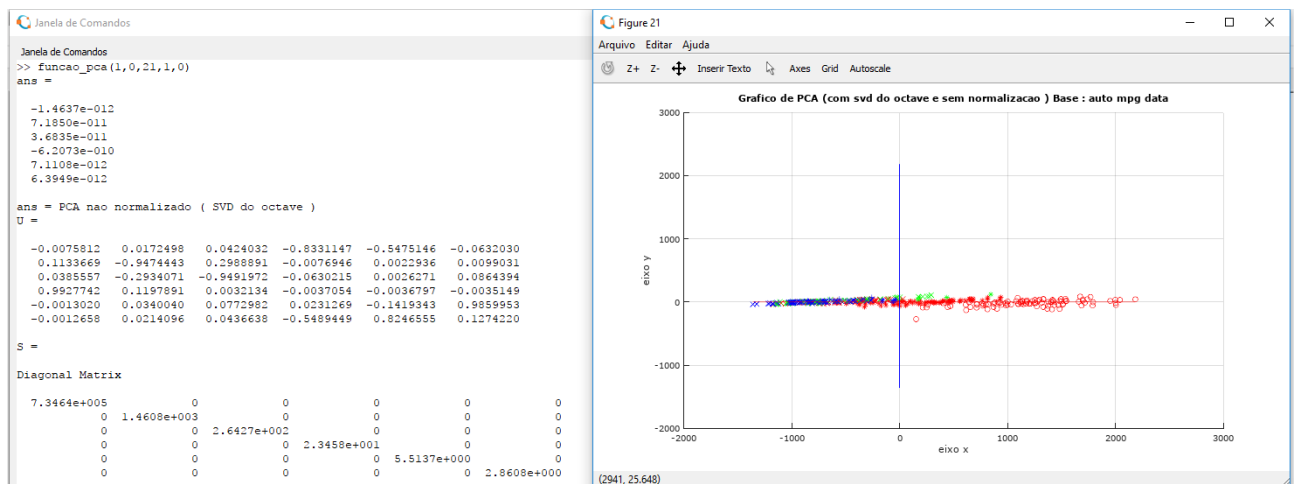


Figura 21: PCA com SVD do Octave e sem utilização da norma para a base auto mpg data.

A Figura 21 também está com indefinição do vetor de tendências.

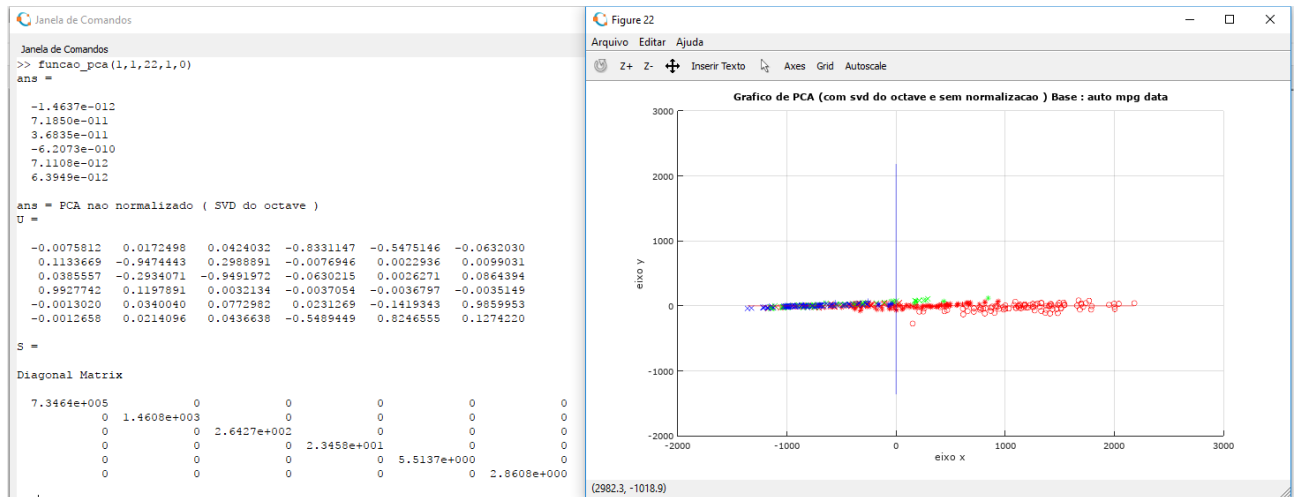


Figura 22: PCA com SVD do Octave e sem utilização da norma para a base auto mpg data.

A Figura 22 é também semelhante às Figuras 21, 20 e 19, com dados compactados e indefinição dos vetores de tendência.

7.2 Base: household power consumption

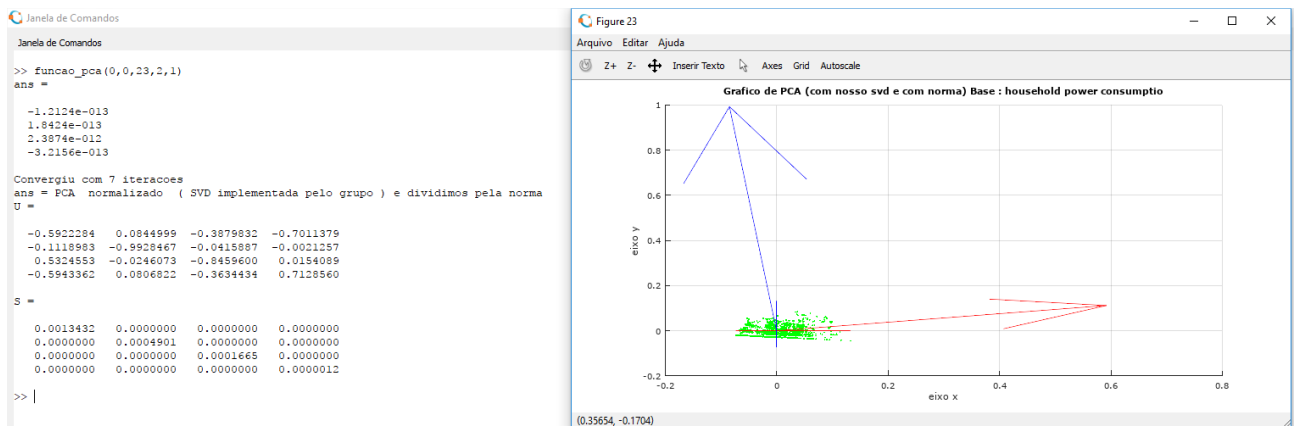


Figura 23: PCA com nosso SVD e utilização da norma para a base household power consumption.

A Figura 23 mostra os dados concentrados na região da origem dos eixos, porém o vetor de tendências está apontando para o primeiro quadrante, demonstrando a relevância dos dados nesse caminho.

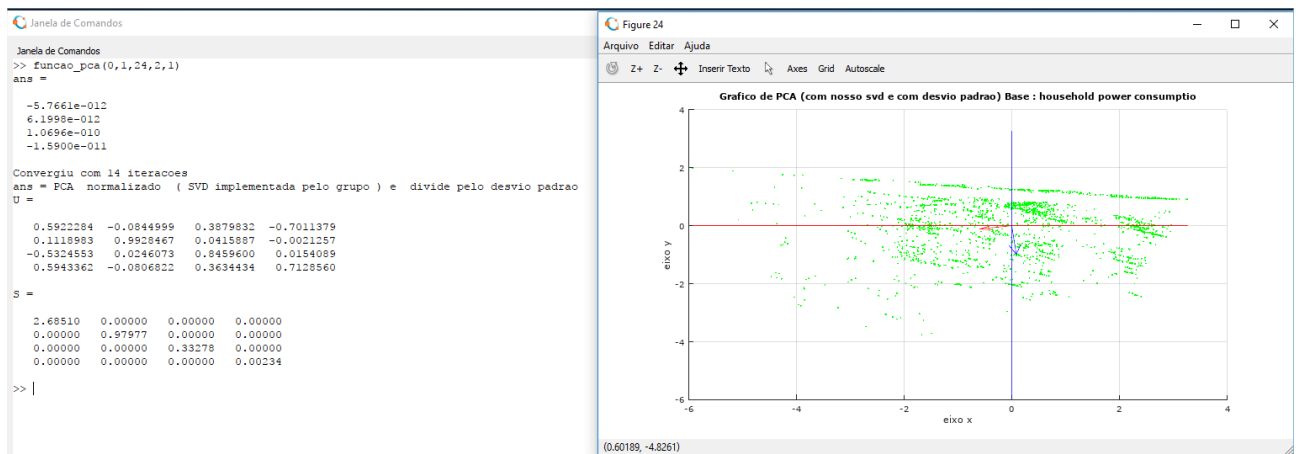


Figura 24: PCA com nosso SVD e utilização do desvio padrão para a base household power consumption.

A Figura 24 apresenta os dados de forma muito espalhada nos quatro quadrantes e com isso fica difícil determinar o vetor de tendências. Porém com muita precisão, pode-se ver que o vetor de tendências está quase sobreposto no eixo X com leve desvio para o terceiro quadrante.

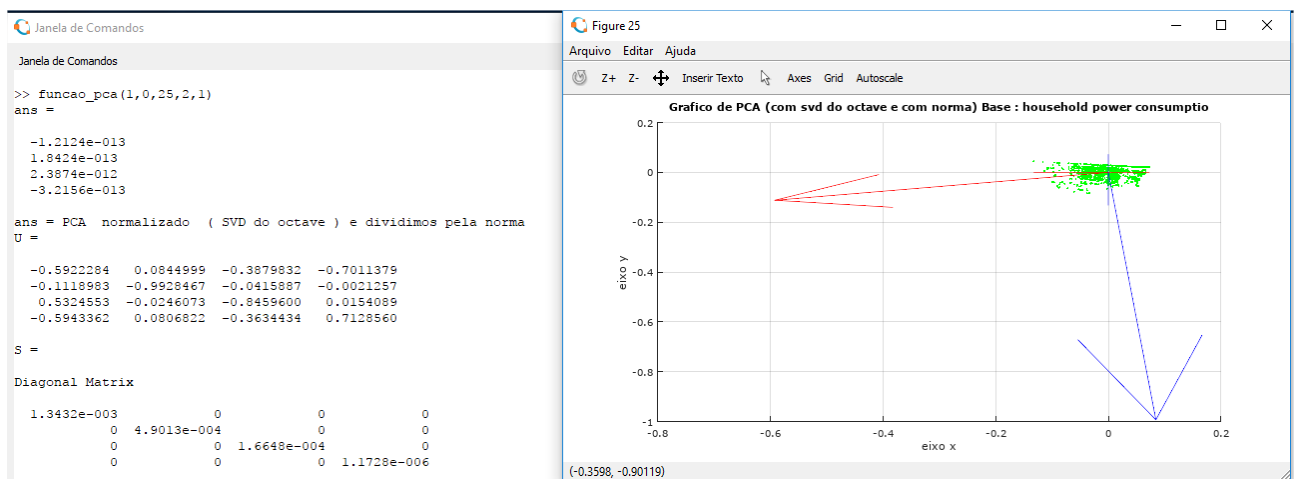


Figura 25: PCA com SVD do Octave e utilização da norma para a base household power consumption.

A Figura 25 mostra que os dados estão bem condensados na região da origem dos eixos e o vetor de tendências está apontando para o terceiro quadrante, demonstrando relevância dos dados nessa área.

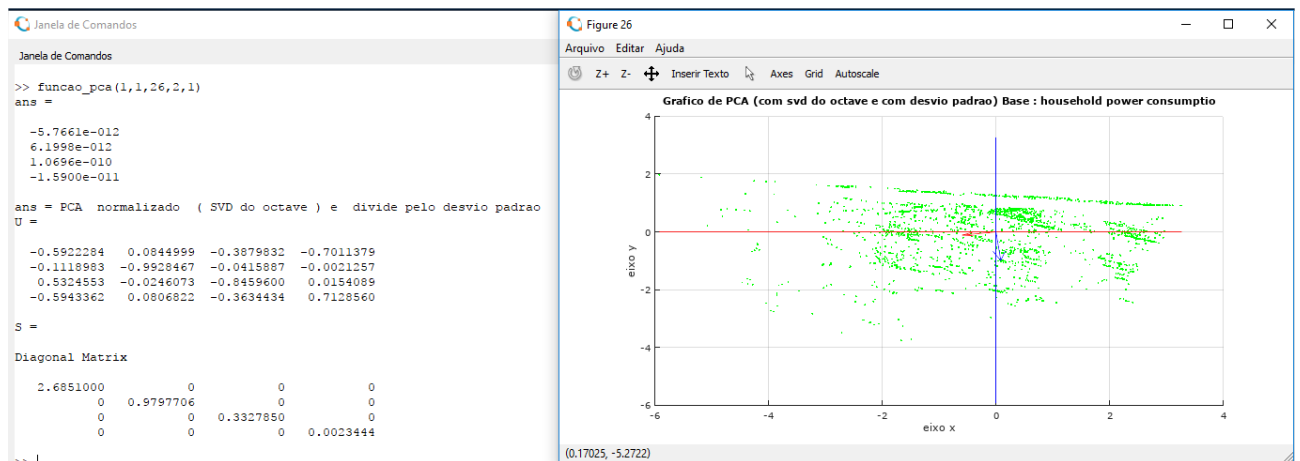


Figura 26: PCA com SVD do Octave e utilização do desvio padrão para a base household power consumption.

A Figura 26 apresenta os dados de forma muito espalhada e com maior proporção no terceiro quadrante. O vetor de tendências aponta para o terceiro quadrante.

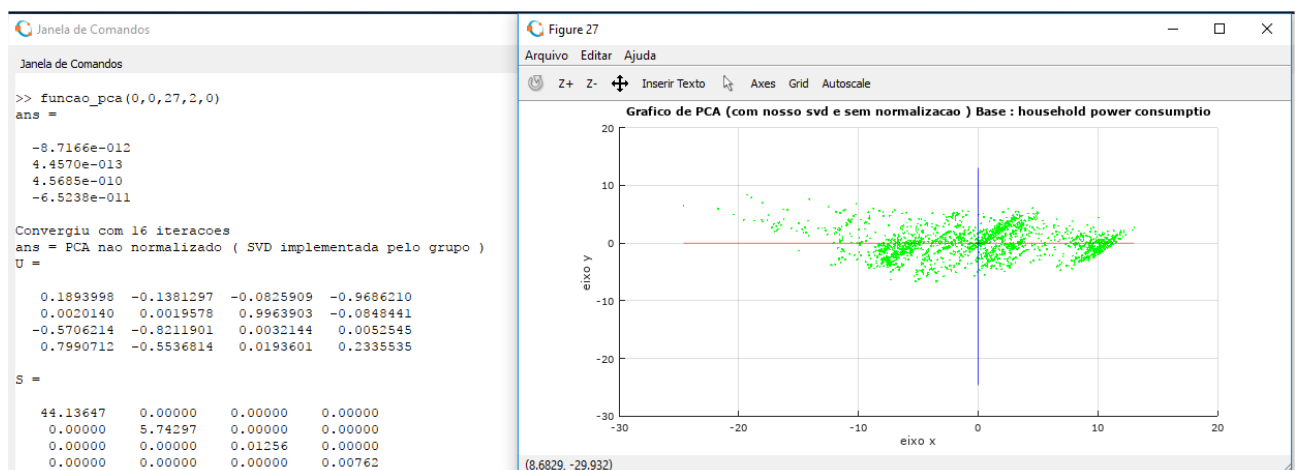


Figura 27: PCA com nosso SVD e sem utilização da norma para a base household power consumption.

A Figura 27 apresenta os dados de forma bem uniforme na origem dos eixos. Porém o vetor de tendências não aparece.

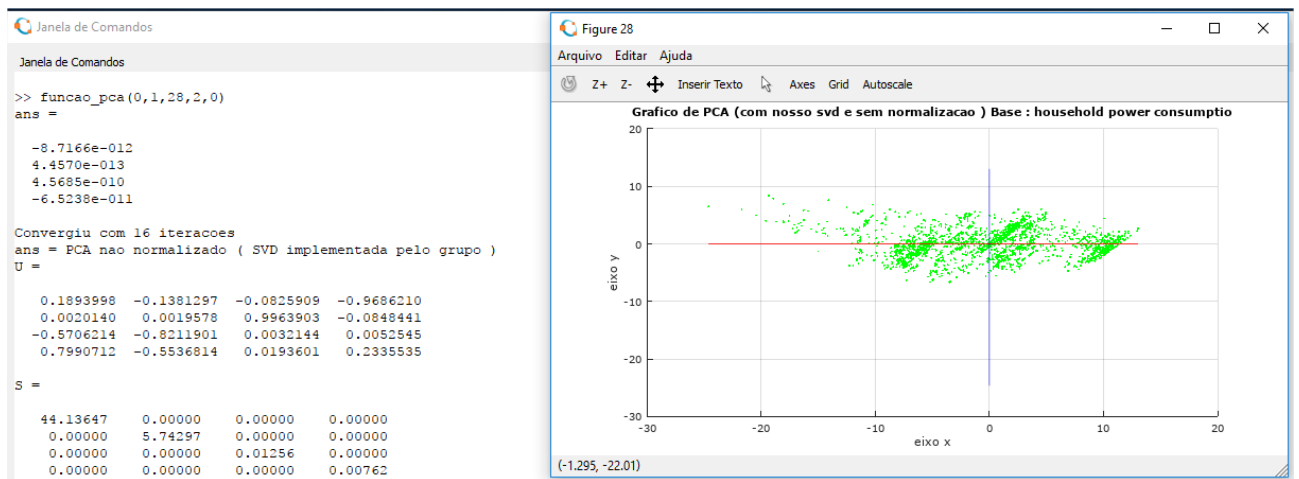


Figura 28: PCA com nosso SVD e sem utilização da norma para a base household power consumption.

Semelhante à Figura 27, a Figura 28 também apresenta os dados de forma espalhada e uniforme, dificultando o aparecimento do vetor de tendência dos dados.

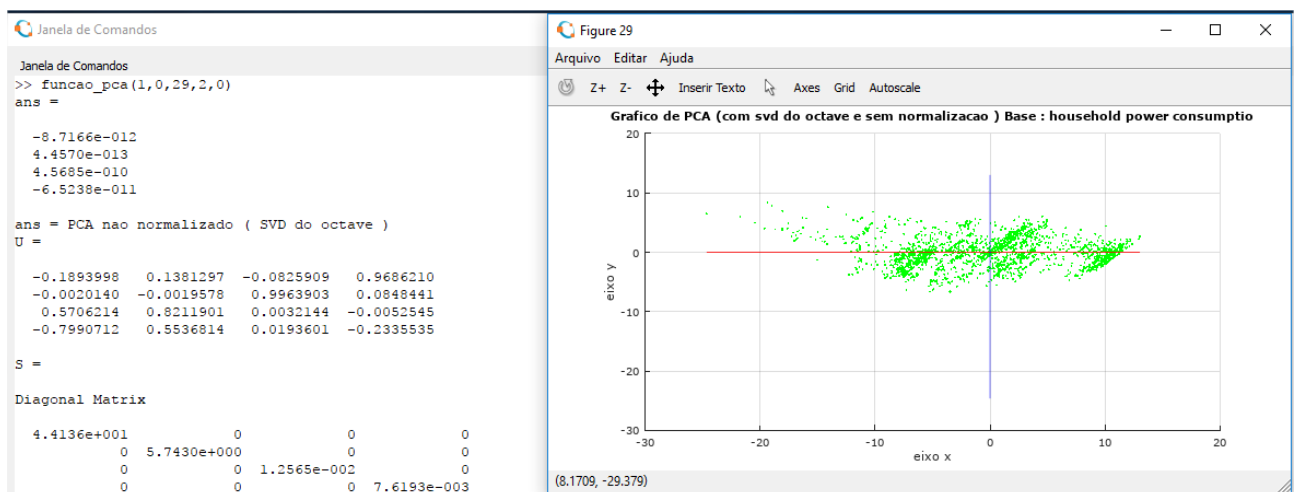


Figura 29: PCA com SVD do Octave e sem utilização da norma para a base household power consumption.

A Figura 29 é também semelhante às figuras 28 e 27, com indefinição do vetor de tendências e com dados espalhados e uniformes.

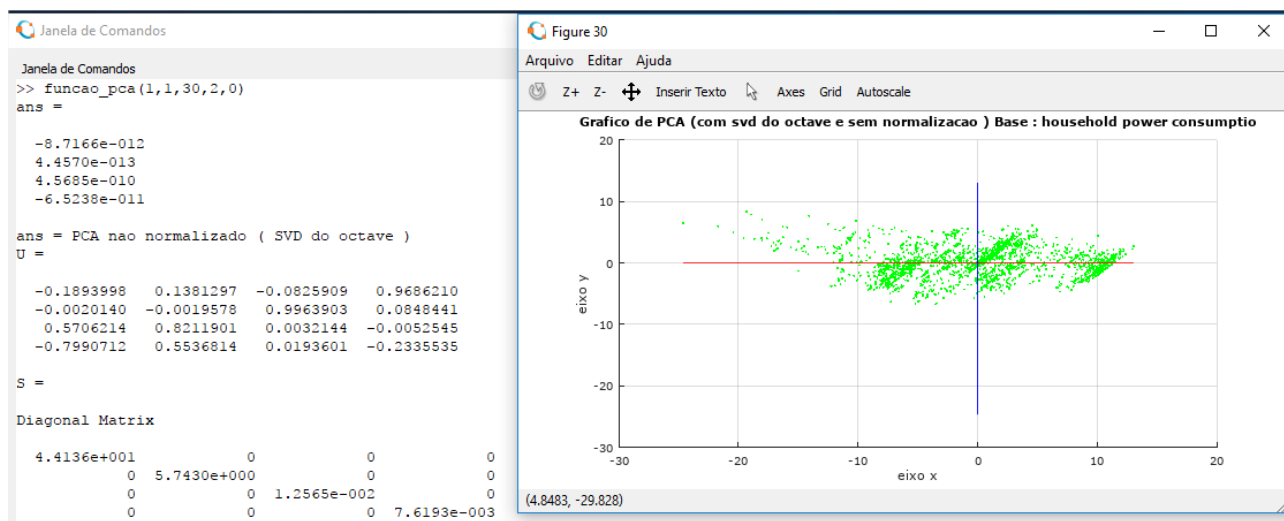


Figura 30: PCA com SVD do Octave e sem utilização da norma para a base household power consumption.

A Figura 30 também apresenta indefinição do vetor de tendências e com dados espalhados e uniformes.

8 Conclusões

Para os bancos de dados aplicados, o método PCA se apresentou mais eficiente para análise.

	Iterações	Chamada de funções		
MQ	163	my_SVD	Desvio Padrão	Normalizar
	251	my_SVD	Norma	Normalizar
	200060	my_SVD	Desvio Padrão	Não normalizar
	200060	my_SVD	Norma	Não normalizar
PCA	7	my_SVD	Desvio Padrão	Normalizar
	14	my_SVD	Norma	Normalizar
	16	my_SVD	Desvio Padrão	Não normalizar
	16	my_SVD	Norma	Não normalizar

Conclui-se também que para ambas as bases que não faz diferença o uso do nosso SVD ou o do Octave, pois ambos apresentam praticamente os mesmos resultados.

Já o uso da Norma nos dados faz compactar/centralizar os dados, tornando-os mais fáceis de serem interpretados. O não uso da norma ou o uso do desvio padrão faz os dados estarem mais dispersos e difíceis de criar um vetor de tendências.

Tanto PCA quanto o MQ apresentam boas visualizações para o vetor de tendências, caracterizando bem os dados representados.

9 Referências

QUARTERONI, A. ; SALERI, F. e GERVASIO, P. Scientific Computing With Matlab and Octave. Italy : Springer, 4ed, 2007

Golub, G. H. and Van Loan, C. F. "The Singular Value Decomposition" and "Unitary Matrices." §2.5.3 and 2.5.6 in Matrix Computations, 3rd ed. Baltimore, MD: Johns Hopkins University Press, pp. 70-71 and 73, 1996.

BURDEN, R. L. e FAIRES, J. D. Análise Numérica. São Paulo: CENCAGE, 8ed, 2011.

10 Anexos – Algoritmo em octave

10.1 Autovetores

```
function [V] = my_autovetores(T,U,lambda)
[linha,coluna] = size(T);
V = zeros(coluna); % matriz com valores zeros
for k=1:coluna
    yy = -(T(1:k-1,1:k-1) - lambda(k)*eye(k-1)) \ T(1:k-1,k);
    y = [yy; 1; zeros(coluna-k,1)];
    V(:,k) = U*y;
end
endfunction
```

10.2 SVD

```
% S- vetor de coluna de valores singulares
% U- Matriz de vetores singular esquerdo
% V- Matriz de vetores singular direito
function [U,S,V] = my_svd(A)
    %retorna a dimensão da matriz, linha=m e coluna=n
    [m,n] = size(A);
    if (m < n)
        error('m deve ser maior ou igual a n');
    end
    ATA = A' * A; % garante que a matriz ATA é quadrada
    %Cálculo dos autovalores
    [lambda,niter,Q,T] = my_qr_iterativo(ATA,1e-12,200065);
    %cálculo dos autovetores
    V = my_autovetores(T,Q,lambda);
    % S é uma matriz com zeros da mesma ordem da matriz A
    S = zeros(size(A));
    % extrai a raiz de cada elemento de lambda e adiciona
    % na diagonal principal da matriz S
    S(1:n,1:n) = diag(sqrt(lambda));
    AV = A*V;
    %Normaliza os dados com a norma
    U= Normalizar_dados(m,n,AV);
endfunction
```

10.3 QR

```
% A matriz A sempre é quadrada
function [lambda,numero_maxima_de_iteracao,U,T] =
my_qr_iterativo(A,erro_sugerido,numero_maxima_de_iteracao)
```

```

        T = A;
        iteracoes = 0;
% retorna o maior valor absoluto entre a coluna 1 até número de coluna (n-1)
        erro = max(max(abs(tril(T,-1))));
% matriz identidade na mesma ordem da matriz A
        U = eye(size(A));
        while (erro > erro_sugerido) && (iteracoes <=
numero_maxima_de_iteracao)
            [Q,R] = qr(T);
            T = R*Q;
            U = U*Q;
            erro = max(max(abs(tril(T,-1))));
            iteracoes++;
        end
        if (iteracoes >= numero_maxima_de_iteracao)
            printf('atingiu o numero maximo de iteracoes : %d ',iteracoes);
            error(', Nao convergiu');
        end
        if (erro <= erro_sugerido)
            printf('Convergiu com %d iteracoes',iteracoes);
        end
        % valores da diagonal da matriz T
        lambda = diag(T);
endfunction

```

10.4 plot line direction

```

function plot_line_direction(X, dir, color)
% desenha o ambiente gráfico com o eixo x e Y
% e plota os dados
pmin = min(X')'; % valores da primeira coluna
pmax = max(X')'; % valores da última coluna
xmin = pmin(1); % busca o menor valor
xmax = pmax(1); % busca o maior valor

hold on;
if dir(1)~=0
    plot ( [xmin,xmax],dir(2)/dir(1)*[xmin,xmax], color);
else
    plot ( dir(1)/dir(2)*[xmin,xmax],[xmin,xmax], color);
end
hold off;

endfunction

```

10.5 gerar gráfico mq

```

function
gerar_grafico_mq(V,H,Dados,Xtudo,SVD,NORMA,NORMALIZAR,base_usada,FIGURA,BASE,m)
%rotação dos eixos para possibilitar a visualização em R 2
Xrot = V * H;
%plota figura com o numero informado no parâmetro FIGURA
figure(FIGURA);
hold on;
if (BASE == 1)
% plotagem do grafico auto mpg data -----
    plot (Xrot(3,:), Xrot(4,:), 'go');
    Xproj = Xrot(1:2,:);
    pmin = min(Xproj)';

```



```

        pmax = max(Xproj')';
% desenha as categorias e subcategorias com cores diferentes, de acordo com a
% informado dadaset que retiramos
    cat1 = find(Dados(8,:) == 1); %pega todas as linhas que na coluna 8 for = 1
    cat2 = find(Dados(8,:) == 2); %pega todas as linhas que na coluna 8 for = 2
    cat3 = find(Dados(8,:) == 3); %pega todas as linhas que na coluna 8 for = 3
% pega todas as linhas que na coluna 8 a categoria = 1 e na coluna 2 forem
% (8,6,4,3)
        z = 1; z2 = 1; z3 = 1; z4 = 1;
        cat11 = zeros(1,1);
        cat12 = zeros(1,1);
        cat13 = zeros(1,1);
        cat14 = zeros(1,1);
        for i=1:length(cat1)
            v = cat1(1,i);
            if (Xtudo(2,v) == 8)
                cat11(1,z) = v;
                z++;
            elseif (Xtudo(2,v) == 6)
                cat12(1,z2) = v;
                z2++;
            elseif (Xtudo(2,v) == 4)
                cat13(1,z3) = v;
                z3++;
            elseif (Xtudo(2,v) == 3)
                cat14(1,z4) = v;
                z4++;
            end
            i++;
        end
% pega todas as linhas que na coluna 8 a categoria = 2 e na coluna 2 forem
% (8,6,4,3)
        z = 1; z2 = 1; z3 = 1; z4 = 1;
        cat21 = zeros(1,1);
        cat22 = zeros(1,1);
        cat23 = zeros(1,1);
        cat24 = zeros(1,1);
        for i=1:length(cat2)
            v = cat2(1,i);
            if (Xtudo(2,v) == 8)
                cat21(1,z) = v;
                z++;
            elseif (Xtudo(2,v) == 6)
                cat22(1,z2) = v;
                z2++;
            elseif (Xtudo(2,v) == 4)
                cat23(1,z3) = v;
                z3++;
            elseif (Xtudo(2,v) == 3)
                cat24(1,z4) = v;
                z4++;
            end
            i++;
        end
% pega todas as linhas que na coluna 8 a categoria = 3 e na coluna 2 forem
% (8,6,4,3)
        z = 1;
        z2 = 1;
        z3 = 1;
        z4 = 1;
        cat31 = zeros(1,1);

```

```

cat32 = zeros(1,1);
cat33 = zeros(1,1);
cat34 = zeros(1,1);
for i=1:length(cat3)
    v = cat3(1,i);
    if (Xtudo(2,v) == 8)
        cat31(1,z) = v;
        z++;
    elseif (Xtudo(2,v) == 6)
        cat32(1,z2) = v;
        z2++;
    elseif (Xtudo(2,v) == 4)
        cat33(1,z3) = v;
        z3++;
    elseif (Xtudo(2,v) == 3)
        cat34(1,z4) = v;
        z4++;
    end
    i++;
end
%plota os dados
if (cat11(1,1)>0)
    plot (Xproj(1,cat11(1,:)), Xproj(2,cat11(1,:)), 'ro');
end
if (cat12(1,1)>0)
    plot (Xproj(1,cat12(1,:)), Xproj(2,cat12(1,:)), 'r*');
end
if (cat13(1,1)>0)
    plot (Xproj(1,cat13(1,:)), Xproj(2,cat13(1,:)), 'rx');
end
if (cat14(1,1)>0)
    plot (Xproj(1,cat14(1,:)), Xproj(2,cat14(1,:)), 'r+');
end
if (cat21(1,1)>0)
    plot (Xproj(1,cat21(1,:)), Xproj(2,cat21(1,:)), 'go');
end
if (cat22(1,1)>0)
    plot (Xproj(1,cat22(1,:)), Xproj(2,cat22(1,:)), 'g*');
end
if (cat23(1,1)>0)
    plot (Xproj(1,cat23(1,:)), Xproj(2,cat23(1,:)), 'gx');
end
if (cat24(1,1)>0)
    plot (Xproj(1,cat24(1,:)), Xproj(2,cat24(1,:)), 'g+');
end
if (cat31(1,1)>0)
    plot (Xproj(1,cat31(1,:)), Xproj(2,cat31(1,:)), 'bo');
end
if (cat32(1,1)>0)
    plot (Xproj(1,cat32(1,:)), Xproj(2,cat32(1,:)), 'b*');
end
if (cat33(1,1)>0)
    plot (Xproj(1,cat33(1,:)), Xproj(2,cat33(1,:)), 'bx');
end
if (cat34(1,1)>0)
    plot (Xproj(1,cat34(1,:)), Xproj(2,cat34(1,:)), 'b+');
end
else
    % plotagem do gráfico household_power_consumptio -----
    % visualização dos dados sobre o hiperplano nas direções 2 e 4
    plot (Xrot(2,:), Xrot(4,:), 'g.');
```

```

        Xproj = Xrot(1:2,:);

        pmin = min(Xproj)';
        pmax = max(Xproj)';

    end
    % calcular o eixos cartesianos e rotaciona no espaço dos eixos principais
    Eixo = V' * eye(m);
    quiver(0,0, Eixo(1,1), Eixo(1,2), 'r');
    quiver(0,0, Eixo(2,1), Eixo(2,2), 'b');
    % desenhar direções principais alinhadas com os eixos cartesianos
    plot_line_direction(Xproj, [1;0], '-r');
    plot_line_direction(Xproj, [0;1], '-b');
    %desenha a legenda, título etc. no gráfico
    xlabel('eixo x');
    ylabel('eixo y');
    titlesvd = ''; titlenorma = '';
    if (SVD == 1)
        titlesvd = ' SVD do octave';
    else
        titlesvd = ' nosso SVD';
    end
    if (NORMA == 1)
        if (NORMALIZAR == 1)
            titlenorma = ' e com desvio padrao)';
        else
            titlenorma = ' e sem normalizacao)';
        end
    else
        if (NORMALIZAR == 1)
            titlenorma = ' e com norma)';
        else
            titlenorma = ' e sem normalizacao)';
        end
    end
    % seta o título do gráfico de acordo com a escolha
    title(strcat('Grafico de M.Q. (com ', titlesvd, titlenorma, base_usada));
    grid;
    axis([pmin(1) pmax(1) pmin(2) pmax(2)]);
    hold off;
    % visualiza os dados sobre o hiperplano nas direções 3 e 4
Endfunction

```

10.6 gerar gráfico pca

```

function
gerar_grafico_pca(V,Xtil,FIGURA,Dados,BASE,Xtudo,SVD,NORMA,NORMALIZAR,base_dados
,m)
% direções principais: cálculo dos auto vetores correspondentes
% aos maiores autovalores da decomposição singular
% multiplicar pela esquerda por V' significa rotaciona as direções
% principais para coincidir com os eixos cartesianos
Xrot = V' * Xtil;
% projeta no R^2
Xproj = Xrot(1:2,:);
    %plota figura com o numero informado no parâmetro FIGURA
    figure(FIGURA);
    if (BASE == 1)
% plotagem do gráfico auto mpg data -----
        hold on;
        pmin = min(Xproj)';

```

```

        pmax = max(Xproj')';
% desenha as categorias e subcategorias com cores diferentes, de acordo com a
% informação do dataset que retiramos
    cat1 = find(Dados(8,:) == 1); %pega todas as linhas que na coluna 8 for = 1
    cat2 = find(Dados(8,:) == 2); %pega todas as linhas que na coluna 8 for = 2
    cat3 = find(Dados(8,:) == 3); %pega todas as linhas que na coluna 8 for = 3
%pega todas as linhas que na coluna 8 a categoria = 1 e na coluna 2 forem
(8,6,4,3)
    z = 1; z2 = 1; z3 = 1; z4 = 1;
    cat11 = zeros(1,1);
    cat12 = zeros(1,1);
    cat13 = zeros(1,1);
    cat14 = zeros(1,1);
    for i=1:length(cat1)
        v = cat1(1,i);
        if (Xtudo(2,v) == 8)
            cat11(1,z) = v;
            z++;
        elseif (Xtudo(2,v) == 6)
            cat12(1,z2) = v;
            z2++;
        elseif (Xtudo(2,v) == 4)
            cat13(1,z3) = v;
            z3++;
        elseif (Xtudo(2,v) == 3)
            cat14(1,z4) = v;
            z4++;
        end
        i++;
    end
%pega todas as linhas que na coluna 8 a categoria = 2 e na coluna 2 forem
(8,6,4,3)
    z = 1; z2 = 1; z3 = 1; z4 = 1;
    cat21 = zeros(1,1);
    cat22 = zeros(1,1);
    cat23 = zeros(1,1);
    cat24 = zeros(1,1);
    for i=1:length(cat2)
        v = cat2(1,i);
        if (Xtudo(2,v) == 8)
            cat21(1,z) = v;
            z++;
        elseif (Xtudo(2,v) == 6)
            cat22(1,z2) = v;
            z2++;
        elseif (Xtudo(2,v) == 4)
            cat23(1,z3) = v;
            z3++;
        elseif (Xtudo(2,v) == 3)
            cat24(1,z4) = v;
            z4++;
        end
        i++;
    end
% pega todas as linhas que na coluna 8 a categoria = 3 e na coluna 2 forem
% (8,6,4,3)
    z = 1; z2 = 1; z3 = 1; z4 = 1;
    cat31 = zeros(1,1);
    cat32 = zeros(1,1);
    cat33 = zeros(1,1);
    cat34 = zeros(1,1);

```

```

for i=1:length(cat3)
    v = cat3(1,i);
    if (Xtudo(2,v) == 8)
        cat31(1,z) = v;
        z++;
    elseif (Xtudo(2,v) == 6)
        cat32(1,z2) = v;
        z2++;
    elseif (Xtudo(2,v) == 4)
        cat33(1,z3) = v;
        z3++;
    elseif (Xtudo(2,v) == 3)
        cat34(1,z4) = v;
        z4++;
    end
    i++;
end
%Plota os dados
if (cat11(1,1)>0)
    plot (Xproj(1,cat11(1,:)), Xproj(2,cat11(1,:)), 'ro');
end
if (cat12(1,1)>0)
    plot (Xproj(1,cat12(1,:)), Xproj(2,cat12(1,:)), 'r*');
end
if (cat13(1,1)>0)
    plot (Xproj(1,cat13(1,:)), Xproj(2,cat13(1,:)), 'rx');
end
if (cat14(1,1)>0)
    plot (Xproj(1,cat14(1,:)), Xproj(2,cat14(1,:)), 'r+');
end
if (cat21(1,1)>0)
    plot (Xproj(1,cat21(1,:)), Xproj(2,cat21(1,:)), 'go');
end
if (cat22(1,1)>0)
    plot (Xproj(1,cat22(1,:)), Xproj(2,cat22(1,:)), 'g*');
end
if (cat23(1,1)>0)
    plot (Xproj(1,cat23(1,:)), Xproj(2,cat23(1,:)), 'gx');
end
if (cat24(1,1)>0)
    plot (Xproj(1,cat24(1,:)), Xproj(2,cat24(1,:)), 'g+');
end
if (cat31(1,1)>0)
    plot (Xproj(1,cat31(1,:)), Xproj(2,cat31(1,:)), 'bo');
end
if (cat32(1,1)>0)
    plot (Xproj(1,cat32(1,:)), Xproj(2,cat32(1,:)), 'b*');
end
if (cat33(1,1)>0)
    plot (Xproj(1,cat33(1,:)), Xproj(2,cat33(1,:)), 'bx');
end
if (cat34(1,1)>0)
    plot (Xproj(1,cat34(1,:)), Xproj(2,cat34(1,:)), 'b+');
end

else
    %plotagem do gráfico household_power_consumptio -----
    hold on;
    pmin = min(Xproj)'; pmax = max(Xproj)';
    %visualização dos dados em R2
    plot (Xproj(1,:), Xproj(2,:), 'g.');
```

```

end
%desenha informações no gráfico
xlabel('eixo x');
ylabel('eixo y');
titlesvd = '';
titlenorma = '';
if (SVD == 1)
    titlesvd = ' svd do octave';
else
    titlesvd = ' nosso svd';
end
if (NORMA == 1)
    if (NORMALIZAR == 1)
        titlenorma = ' e com desvio padrao';
    else
        titlenorma = ' e sem normalizacao );
    end
else
    if (NORMALIZAR == 1)
        titlenorma = ' e com norma);
    else
        titlenorma = ' e sem normalizacao );
    end
end
title(strcat('Grafico de PCA (com ',titlesvd,
titlenorma,base_dados));
grid;
axis([pmin(1) pmax(1) pmin(2) pmax(2)]);
hold off;
% calcular eixos cartesianos e rotaciona no espaço dos eixos principais
Eixo = V' * eye(m);
hold on
quiver(0,0, Eixo(1,1), Eixo(1,2), 'r');
quiver(0,0, Eixo(2,1), Eixo(2,2), 'b');
hold off
% desenha as direções principais e alinha com os eixos cartesianos
plot_line_direction(Xproj, [1;0], '-r');
plot_line_direction(Xproj, [0;1], '-b');
endfunction

```

10.7 PCA

```

% Entrada para parâmetros:
% SVD = 1 (calcula com svd do octave)
% SVD diferente de um (calcula com nossa svd implementada)
% NORMA = 1(normaliza pela divisão com a norma)
% NORMA diferente de um (calcula pelo desvio padrão)
% FIGURA: gera a figura com o número especificado.
% NORMALIZAR=1 normaliza os dados
% NORMALIZAR diferente de 1 não normaliza
% BASE =1 usa a base auto-mpg.data.txt
% BASE diferente de 1 usa a base household_power_consumptio.txt

function funcao_pca(SVD,NORMA,FIGURA,BASE,NORMALIZAR)

[base_dados,Dados,X,Xtudo,m,n,Media,variancia] = carrega_base(BASE);
if (NORMALIZAR == 1)

```

```

        if(NORMA==1)
            mensagem3 = ' e divide pelo desvio padrao ';
        else
            mensagem3 = ' e dividimos pela norma ';
        end
        variancia=norma_desvio_padrao(NORMA,m,n,variancia);
        mensagem1 = 'PCA normalizado (';
    else
        mensagem1 = 'PCA nao normalizado (';
        mensagem3 = ' ';
    end
    % verificar se os dados estao centralizados
    sum(variancia,2)
    % calcula a matriz de covariancia
    Sx = variancia * variancia' / (n-1);
    % SVD da matriz de covariancia
    % verifica o parametro SVD para calcular com a opcao informada
    if (SVD == 1)
        mensagem2 = ' SVD do octave )';
        [U,S,V] = svd(Sx);
    else
        mensagem2 = ' SVD implementada pelo grupo )';
        [U,S,V] = my_svd(Sx);
        %corrige rotacao do grafico quando necessario
        V = V * -1;
    end
    strcat (mensagem1, mensagem2,mensagem3)
    U
    S
    gerar_grafico_pca(V,variancia,FIGURA,Dados,BASE,Xtudo,SVD,NORMA,NORMALIZAR,base_
    dados,m);
endfunction

```

10.8 MQ

```

%Entrada para parâmetros:
% SVD = 1 (calcula com svd do octave)
% SVD diferente de um (calcula com nossa svd implementada)
% NORMA = 1(normaliza e divide com a norma)
% NORMA diferente de um (calcula pelo desvio padrão)
% FIGURA: gera a figura com o número especificado.
% BASE=1 usa a base auto mpg data
% BASE diferente de um usa a base household power consumptio
% NORMALIZAR=1 normaliza os dados
% NORMALIZAR diferente de 1 não normaliza
% BASE =1 usa a base auto-mpg.data.txt
% BASE diferente de 1 usa a base household_power_consumptio.txt

function funcao_mq(SVD,NORMA,FIGURA,BASE,NORMALIZAR)
[base_dados,Dados,X,Xtudo,m,n,Media,variancia] = carrega_base(BASE);
if (NORMALIZAR == 1)
    if(NORMA==1)
        mensagem3 = ' e divide pelo desvio padrao ';
    else
        mensagem3 = ' e dividimos pela norma ';
    end
    variancia=norma_desvio_padrao(NORMA,m,n,variancia);
    mensagem1 = 'MQ normalizado (';
else
    mensagem1 = 'MQ nao normalizado (';

```

```

    mensagem3 = ' ';
end
%cálculo de M(A) usando traço dos dados centralizados
Ma =(trace(variancia*variancia')*eye(m))-(variancia*variancia');

% SVD da matriz de covariancia
% verifica o parametro SVD para calcular com a opcao informada
if (SVD == 1)
    [U,S,V] = svd(Ma);
    mensagem2 = ' SVD do octave )';
else
    [U,S,V] = my_svd(Ma);
    mensagem2 = ' SVD implementada pelo grupo )';
end
strcat (mensagem1, mensagem2,mensagem3)
U
S
V
% projeções dos pontos em R4 no hiperplano
% utilizado a média da base de dados e o vetor normal (autovetor ligado ao menor
autovalor)
% calculo: dado centralizado - (dado centralizado, normal) * (normal)
% o comando dot eh o produto interno

for i=1:n
    H(:,i) = variancia(:,i) - dot(variancia(:,i),V(:,m))*V(:,m);
end
gerar_grafico_mq(V,H,Dados,Xtudo,SVD,NORMA,NORMALIZAR,base_dados,FIGURA,BASE,m);
endfunction

```

10.9 Norma ou desvio padrão

```

% NORMA=1 divide pelo desvio padrão
% NORMA diferente de um divide pela norma
function [variancia]=norma_desvio_padrao(NORMA,m,n,variancia)
    for i=1:m
        %verifica o parâmetro NORMA para calcular com a opção informada
        if (NORMA == 1)
            %divide pelo desvio padrão
            variancia(i,:) = variancia(i,:) / sqrt(variancia(i,:)*variancia(i,:)'./(n-1));
        else
            %dividimos pela norma
            variancia(i,:) = variancia(i,:) / norm(variancia(i,:));
        end
    end
end
endfunction

```

10.10 Normalizar dados

```

% pré-condição:
% A matriz AV pode estar com valores de -10000 a 10000
% Pos-condicao:
% resulta numa matriz U com cada elemento
% estando entre -1 e 1, ou seja, normalizado
function [U] = Normalizar_dados(m,n,AV)
    U = eye(m,m); % matriz identidade
    %Normaliza os dados com a norma
    for i=1:n
        U(:,i) = AV(:,i) / norm(AV(:,i));
    end
end

```



```

for k=n+1:m
    soma = 0;
    for j=1:k-1
        soma = soma + ( U(:,k)'*U(:,j) ) / ( U(:,j)'*U(:,j) ) * U(:,j);
    end

    U(:,k) = U(:,k) - soma;
    U(:,k) = U(:,k) / norm(U(:,k));

end
endfunction

```

10.11 Carregar base

```

function [base_dados,Dados,X,Xtudo,m,n,Media,variancia] = carrega_base(BASE)

if (BASE == 1) % pode ser usado a mesma funcao para duas bases
    %armazena os dados sobre os carros
    Dados = load('-ascii','auto-mpg.data.txt');
    base_dados = ' Base : auto mpg data ';
    % estes dados sao divididos em tres classes (A última linha e a
    % segunda) 8 e 2
    % remove as variaveis 8 e 2(classificacao)
    X = Dados([1,3:end-1],:);
else
    %armazena os dados sobre os ibge
    Dados = load('-ascii','household_power_consumptio.txt');
    base_dados = ' Base : household power consumptio ';
    % pega as primeiras 4 variaveis e descartar valores 0
    X = Dados(1:4,:);
end
%armazena matriz auxiliar para a subcategoria
Xtudo = Dados(1:end,:);

[m,n] = size(X);

% calcula as medias
Media = diag(sum(X,2)/size(X,2)) * ones(size(X));

% centraliza dados (variancia)
variancia = X - Media;

endfunction

```

10.12 Comparar SVD Octave X SVD Alternativo

```

%-----%
% Comparativo SVD %
%-----%
% Gera matriz de valores aleatórios com tamanho entre tam1 e tam2 %
% e compara os resultados das execuções das seguintes funções: %
% svd - função nativa do Octave %
% my_svd - função desenvolvida pela equipe, possui parâmetros para %
% controle do número máximo de iterações e da tolerância ao erro. %

```

```

%-----%

function resultados = compara_svd(tam1=3, tam2=tam1, passo=1)

    resultados = zeros(1, 8);
    k=1;
    for j=3:9
        resultados(k, j-1) = 10^(-j*passo)
    end;
    for i=tam1:tam2
        A=rand(i,i);
        [U_Nativo, S_Nativo, V_Nativo] = svd(A);
        k = k+1;
        resultados(k, 1) = i; %tamanho da matriz

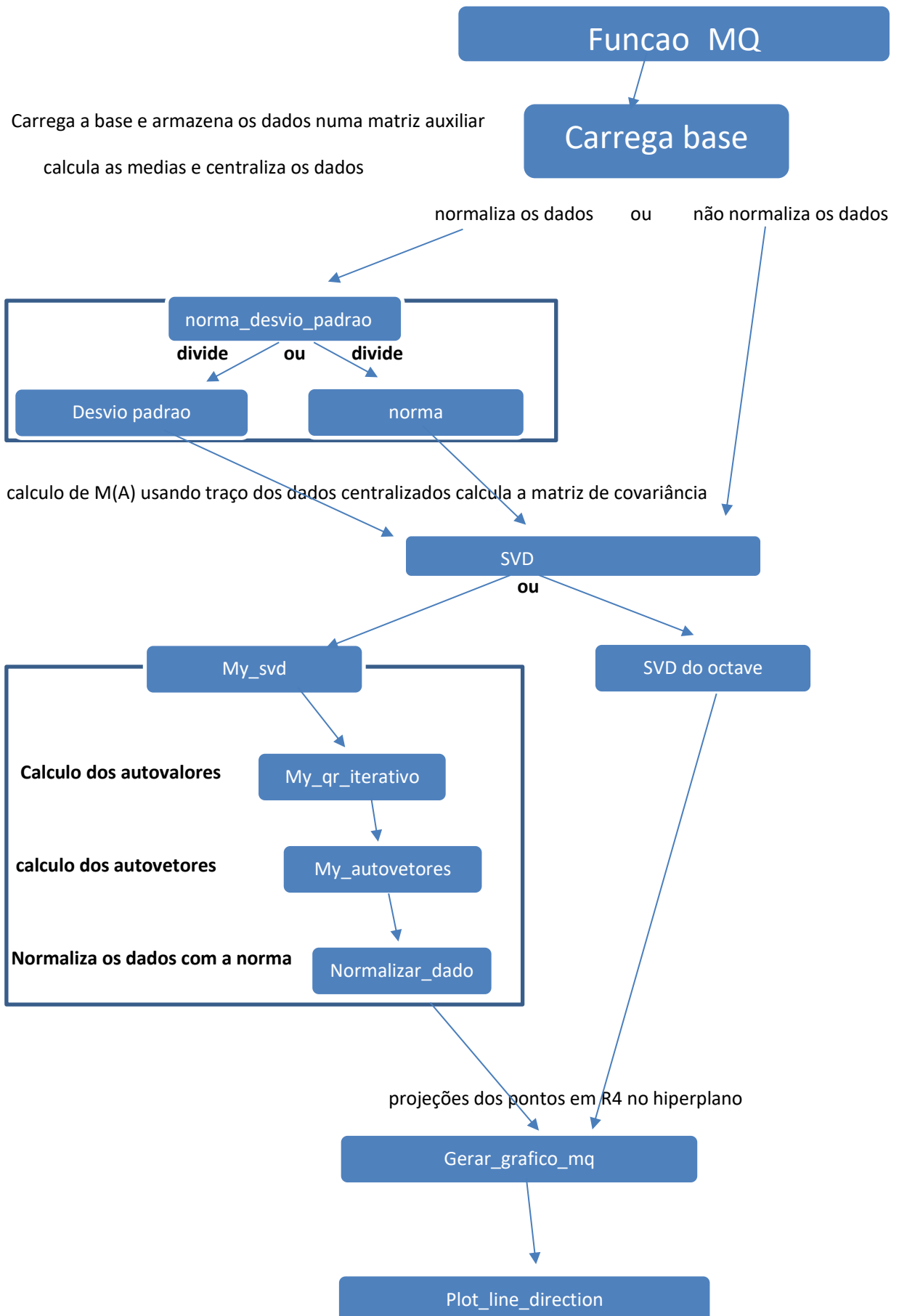
        for j=3:9
            tolerancia = 10^(-j*passo)
            [U_Alternativo, S_Alternativo, V_Alternativo] = my_svd(A, tolerancia);
            resultados(k, j-1) = sum(sum(abs(S_Nativo - S_Alternativo)));

        end;

    end;
    csvwrite("compara_svd.csv", resultados);
endfunction

```

11 Chamadas de funções implementada no MQ



12 Chamadas de funções implementada no PCA

