

1. Octave 和Matlab及其相似，只不过是开源的。

2. ^是指数，xor() 是异或。

3. PS1()改变提示符。

```
1. octave:1> PS1('#')
2. #
```

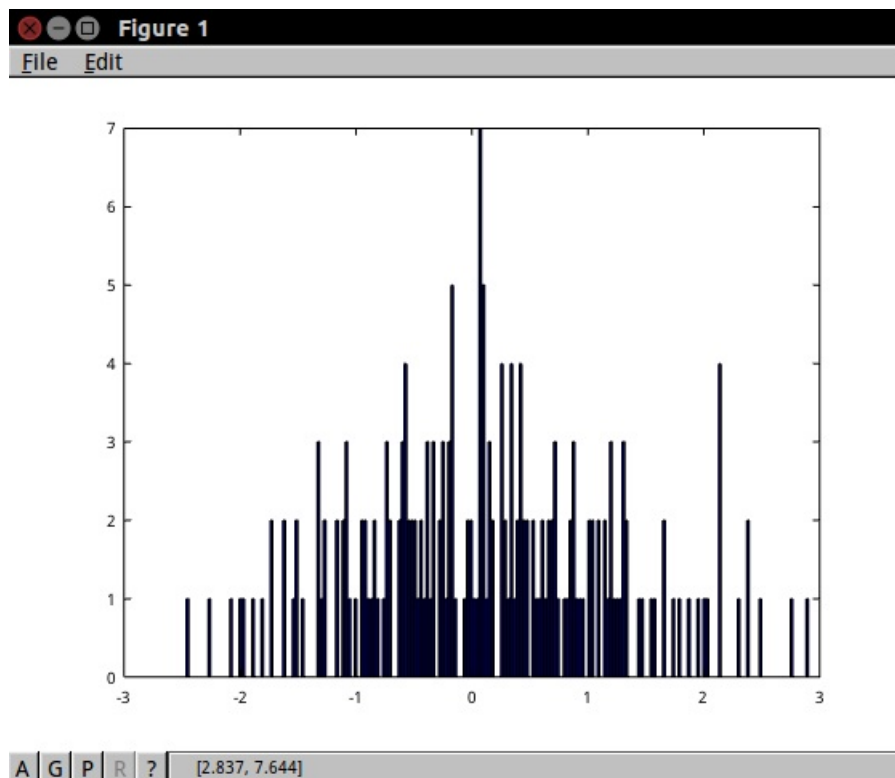
4. disp(sprintf('value: %f', a)).

```
1. #a = pi;
2. #disp(sprintf('value: %f', a))
3. value: 3.141593
4. #
```

5. rand(), randn(). hist(). eye().

```
1. octave:1> X = randn(1, 200);
2. octave:2> hist(X, 200)
3. octave:3>
```

可见为正态分布：



6. size(), length()-最大维度的大小。

```
1. >>>A = [1 2; 3 4; 5 6]
2. A =
3.
4.     1     2
5.     3     4
6.     5     6
7.
8. >>>size(A)
9. ans =
10.
11.     3     2
12.
13. >>>length(A)
14. ans = 3
```

7. load():

dataX文件：

```
dataX (~/.la-Ubuntu) - gedit
+ 打开 保存 撤消
dataX x
1 2011
2 2012
3 2013
4 2014
5 2015|
```

dataY文件：

```
dataY (~/.la-Ubuntu) - gedit
+ 打开 保存 撤消
dataY x
1.3
5.6
34.56
67.1
123.5|
```

```
1. octave:7> load dataX
2. octave:8> dataX
3. dataX =
4.
5.      1  2011
6.      2  2012
7.      3  2013
8.      4  2014
9.      5  2015
10.
11. octave:9> load('dataY');
12. octave:10> dataY
13. dataY =
14.
15.      1.3000
16.      5.6000
17.     34.5600
18.     67.1000
19.    123.5000
20.
21. octave:11>
```

8. who, whos. 查看工作区间中变量：

```

1. octave:11> who
2. Variables in the current scope:
3.
4. X      ans      dataX  dataY
5.
6. octave:12> whos
7. Variables in the current scope:
8.
9.      Attr Name      Size      Bytes  Class
10.     ====  =====  =====  =====
11.          X          1x200        1600  double
12.         ans          1x8           8    char
13.        dataX         5x2          80    double
14.        dataY         5x1          40    double
15.
16. Total is 223 elements using 1728 bytes
17.
18. octave:13>

```

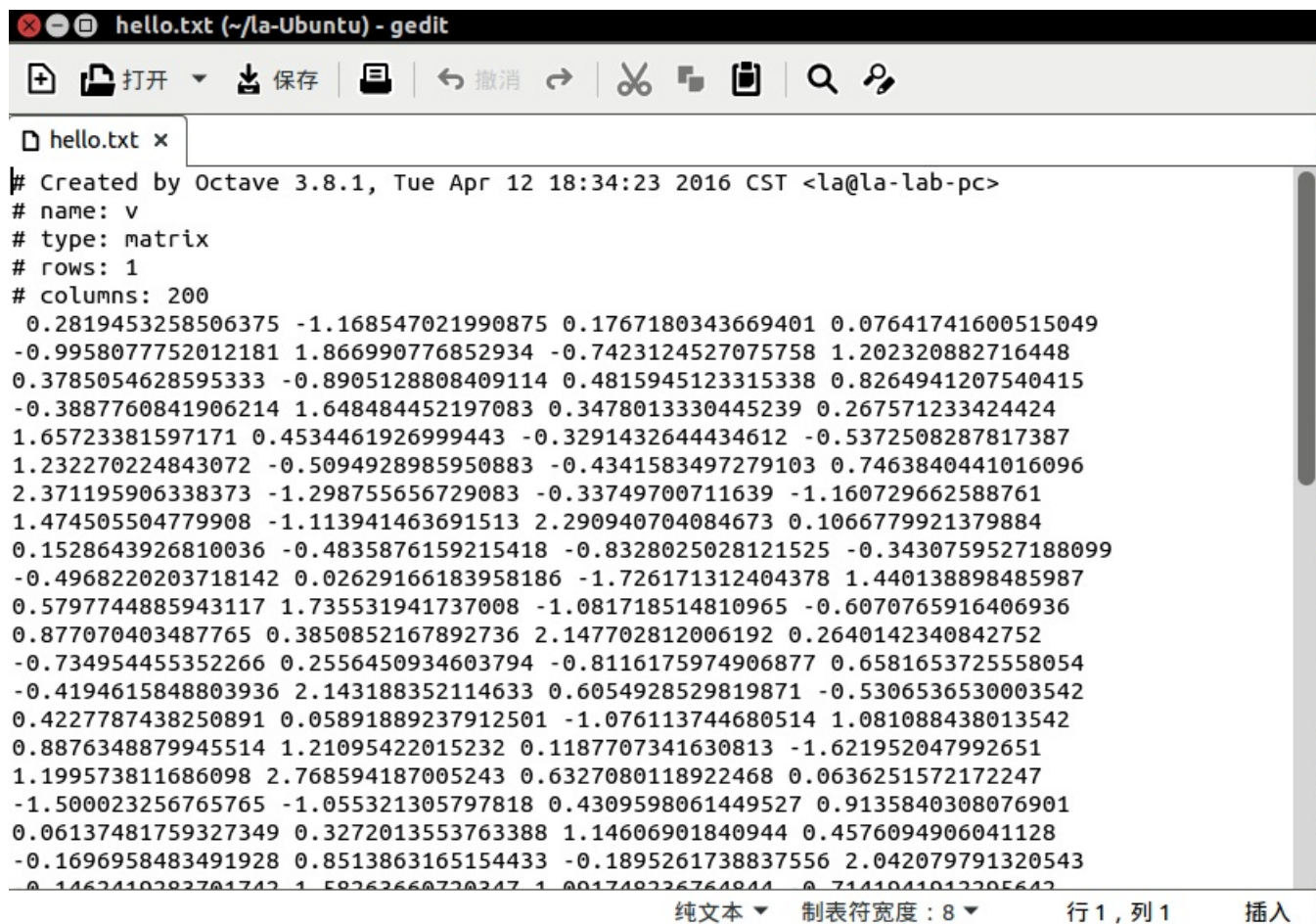
9. save().

```

1. octave:13> v = X;
2. octave:14> save hello.txt v
3. octave:15>

```

hello.mat文件：



```

hello.txt x
# Created by Octave 3.8.1, Tue Apr 12 18:34:23 2016 CST <la@la-lab-pc>
# name: v
# type: matrix
# rows: 1
# columns: 200
0.2819453258506375 -1.168547021990875 0.1767180343669401 0.07641741600515049
-0.9958077752012181 1.866990776852934 -0.7423124527075758 1.202320882716448
0.3785054628595333 -0.8905128808409114 0.4815945123315338 0.8264941207540415
-0.3887760841906214 1.648484452197083 0.3478013330445239 0.267571233424424
1.65723381597171 0.4534461926999443 -0.3291432644434612 -0.5372508287817387
1.232270224843072 -0.5094928985950883 -0.4341583497279103 0.7463840441016096
2.371195906338373 -1.298755656729083 -0.33749700711639 -1.160729662588761
1.474505504779908 -1.113941463691513 2.290940704084673 0.1066779921379884
0.1528643926810036 -0.4835876159215418 -0.8328025028121525 -0.3430759527188099
-0.4968220203718142 0.02629166183958186 -1.726171312404378 1.440138898485987
0.5797744885943117 1.735531941737008 -1.081718514810965 -0.6070765916406936
0.877070403487765 0.3850852167892736 2.147702812006192 0.2640142340842752
-0.734954455352266 0.2556450934603794 -0.8116175974906877 0.6581653725558054
-0.4194615848803936 2.143188352114633 0.6054928529819871 -0.5306536530003542
0.4227787438250891 0.05891889237912501 -1.076113744680514 1.081088438013542
0.8876348879945514 1.21095422015232 0.1187707341630813 -1.621952047992651
1.199573811686098 2.768594187005243 0.6327080118922468 0.0636251572172247
-1.500023256765765 -1.055321305797818 0.4309598061449527 0.9135840308076901
0.06137481759327349 0.3272013553763388 1.14606901840944 0.4576094906041128
-0.1696958483491928 0.8513863165154433 -0.1895261738837556 2.042079791320543
0.1462410282701742 1.58262660720247 0.001748226764844 0.7141041012205642

```

纯文本 ▾ 制表符宽度：8 ▾ 行 1, 列 1 插入

10.

```

1. octave:17> A = rand(3, 4)
2. A =
3.
4.      0.902224      0.517781      0.578266      0.033452
5.      0.160171      0.189596      0.302388      0.079403
6.      0.415101      0.704883      0.734259      0.016870
7.
8. octave:18> A([1 3], :)
9. ans =
10.
11.      0.902224      0.517781      0.578266      0.033452
12.      0.415101      0.704883      0.734259      0.016870
13.
14. octave:19> A(:, 2)
15. ans =
16.
17.      0.51778
18.      0.18960
19.      0.70488
20.
21. octave:20>

```

11. A(:) 把A 中所有元素组织成列向量。

```

1. octave:20> A
2. A =
3.
4.      0.902224      0.517781      0.578266      0.033452
5.      0.160171      0.189596      0.302388      0.079403
6.      0.415101      0.704883      0.734259      0.016870
7.
8. octave:21> A(:)
9. ans =
10.
11.      0.902224
12.      0.160171
13.      0.415101
14.      0.517781
15.      0.189596
16.      0.704883
17.      0.578266
18.      0.302388
19.      0.734259
20.      0.033452
21.      0.079403
22.      0.016870

```

12.

```

1. octave:22> A = [1 2; 3 4; 5 6]
2. A =
3.
4.      1      2
5.      3      4
6.      5      6
7.
8. octave:23> B = [11 12; 13 14; 15 16]
9. B =
10.
11.     11     12
12.     13     14
13.     15     16
14.
15. octave:24> C = [1 1; 2 2]
16. C =
17.
18.      1      1
19.      2      2
20.
21. octave:25> A .* B
22. ans =
23.
24.     11     24

```

```

25.      39    56
26.      75    96
27.
28. octave:26> A .^ 2
29. ans =
30.
31.      1     4
32.      9    16
33.     25    36
34.
35. octave:27> v = [1; 2; 3]
36. v =
37.
38.      1
39.      2
40.      3
41.
42. octave:28> 1 ./ v
43. ans =
44.
45.      1.00000
46.      0.50000
47.      0.33333
48.
49. octave:29> 1 ./ A
50. ans =
51.
52.      1.00000    0.50000
53.      0.33333    0.25000
54.      0.20000    0.16667
55.
56. octave:30> log(v)
57. ans =
58.
59.      0.00000
60.      0.69315
61.      1.09861
62.
63. octave:31> exp(v)
64. ans =
65.
66.      2.7183
67.      7.3891
68.     20.0855
69.
70. octave:32> abs(v)
71. ans =
72.
73.      1
74.      2
75.      3
76.
77. octave:33> abs([-1;-2;-3])
78. ans =
79.
80.      1
81.      2
82.      3
83.
84. octave:34> -v
85. ans =
86.
87.     -1
88.     -2
89.     -3
90.
91. octave:35> v + ones(length(v), 1)
92. ans =
93.
94.      2
95.      3
96.      4
97.
98. octave:36> length(v)
99. ans = 3

```

```

100. octave:37> ones(3, 1)
101. ans =
102.
103.     1
104.     1
105.     1
106.
107. octave:38> A
108. A =
109.
110.     1     2
111.     3     4
112.     5     6
113.
114. octave:39> A'
115. ans =
116.
117.     1     3     5
118.     2     4     6
119.
120. octave:40> (A')'
121. ans =
122.
123.     1     2
124.     3     4
125.     5     6
126.
127. octave:41> a = [1 15 2 0.5]
128. a =
129.
130.     1.00000     15.00000     2.00000     0.50000
131.
132. octave:42> val = max(a)
133. val = 15
134. octave:43> [val, ind] = max(a)
135. val = 15
136. ind = 2
137. octave:44> max(A)
138. ans =
139.
140.     5     6
141.
142. octave:45> a
143. a =
144.
145.     1.00000     15.00000     2.00000     0.50000
146.
147. octave:46> a < 3
148. ans =
149.
150.     1     0     1     1
151.
152. octave:47> find(a < 3)
153. ans =
154.
155.     1     3     4
156.
157. octave:48> magic(a)
158. octave:49> magic(3)
159. ans =
160.
161.     8     1     6
162.     3     5     7
163.     4     9     2

```

```

1. octave:60> a
2. a =
3.
4.      1.00000    15.00000    2.00000    0.50000
5.
6. octave:61> prod(a)
7. ans = 15
8. octave:62> A
9. A =
10.
11.     1     2
12.     3     4
13.     5     6
14.
15. octave:63> prod(A)
16. ans =
17.
18.     15     48
19.
20. octave:64> floor(a)
21. ans =
22.
23.     1    15     2     0
24.
25. octave:65> ceil(a)
26. ans =
27.
28.     1    15     2     1
29.
30. octave:66> round(a)
31. ans =
32.
33.     1    15     2     1

```

14.

```

1. octave:68> A = magic(3)
2. A =
3.
4.     8     1     6
5.     3     5     7
6.     4     9     2
7.
8. octave:69> max(A, [], 1)
9. ans =
10.
11.     8     9     7
12.
13. octave:70> max(A, [], 2)
14. ans =
15.
16.     8
17.     7
18.     9
19.
20. octave:71> max(A(:))
21. ans = 9
22. octave:72> sum(A)
23. ans =
24.
25.    15    15    15
26.
27. octave:73> sum(A, 1)
28. ans =
29.
30.    15    15    15
31.
32. octave:74> sum(A, 2)
33. ans =
34.
35.    15
36.    15
37.    15

```

15.

```
1. octave:77> A = eye(3)
2. A =
3.
4. Diagonal Matrix
5.
6.      1   0   0
7.      0   1   0
8.      0   0   1
9.
10. octave:78> flipud(A)
11. ans =
12.
13. Permutation Matrix
14.
15.      0   0   1
16.      0   1   0
17.      1   0   0
```

16. 绘图函数。

```
1. octave:82> t = [0:0.01:0.98];
2. octave:83> y1 = sin(2*pi*4*t);
3. octave:84> plot(t, y1);
4. octave:85> y2 = cos(2*pi*4*t);
5. octave:86> hold on
6. octave:87> plot(t, y2, 'r');
7. octave:88> xlabel('time');
8. octave:89> ylabel('value');
9. octave:90> legend('sin', 'cos');
10. octave:91> title("LIAO's plot");
11. octave:92> print -dpng 'LIAOPlot.png'
12. warning: print.m: epstool binary is not available.
13. Some output formats are not available.
14. warning: print.m: fig2dev binary is not available.
15. Some output formats are not available.
16. octave:93>
```