

Demo

- Create a fake TFIDF matrix with a strong concept
- Plot the matrix on a two term axis
- Perform SVD decomposition
- Plot the new axes
- Reduce the dimensionality of SVD
- Plot the new axes

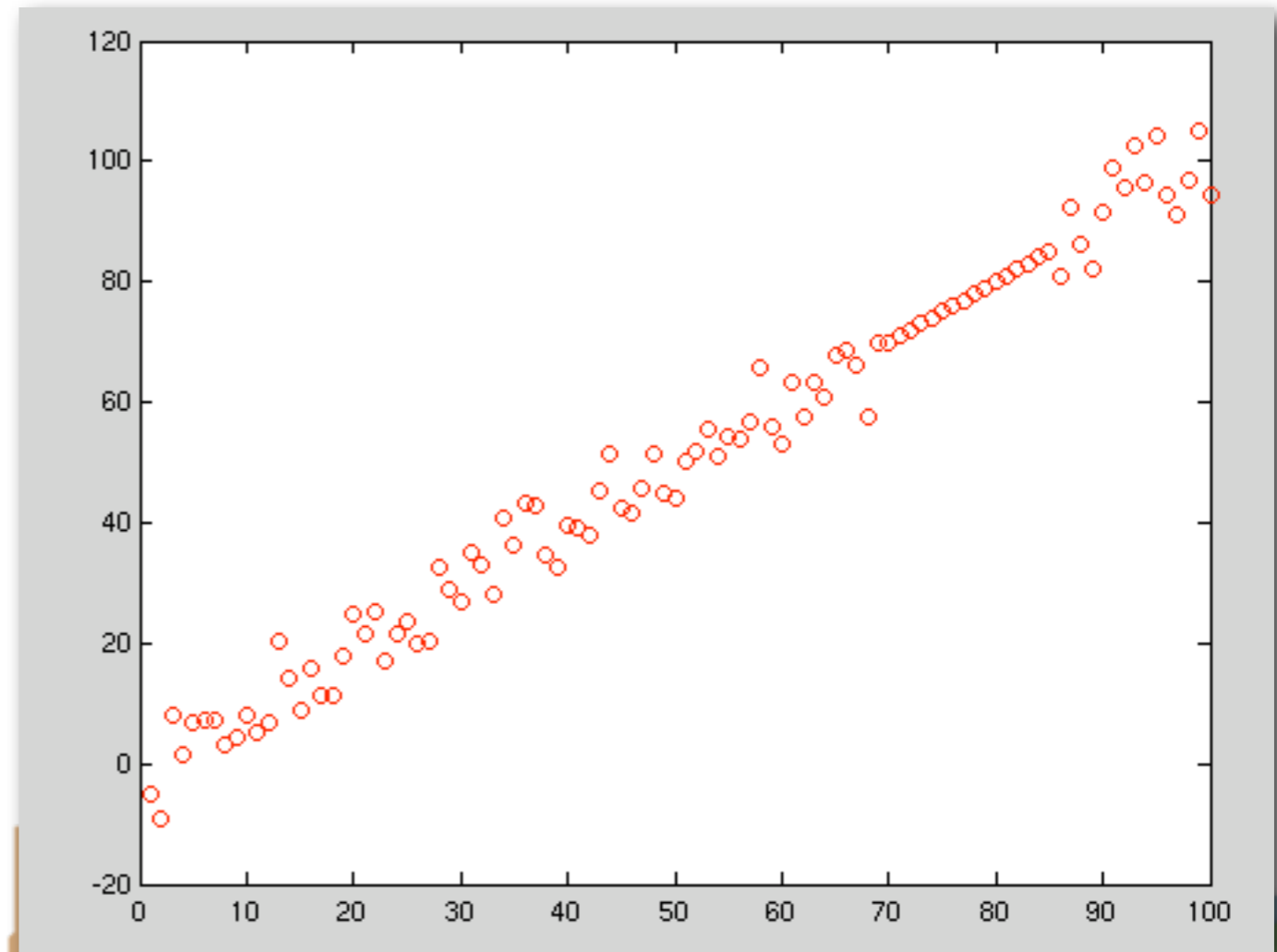


Demo

- Create a fake TFIDF matrix with a strong concept
- Plot the matrix on a two term axis

```
>> x = [1:100];  
>> y = random('norm',0,5,1,100);  
>> y = x + y;  
>> y(70:85) = [70:85];  
>> plot(x,y,'ro')
```

```
>> size(x)  
  
ans =  
  
     1    100  
  
>> size(y)  
  
ans =  
  
     1    100
```



Demo

- Perform SVD decomposition

```
>> C = [x;y];  
>> size(C)  
  
ans =  
  
     2    100  
  
>> [U S V] = svd(C);  
>> size(U)  
  
ans =  
  
     2     2
```

```
>> size(S)  
  
ans =  
  
     2    100  
  
>> S(1:2,1:2)  
  
ans =  
  
 822.6330     0  
         0 30.2548  
  
>> size(V)  
  
ans =  
  
 100    100
```

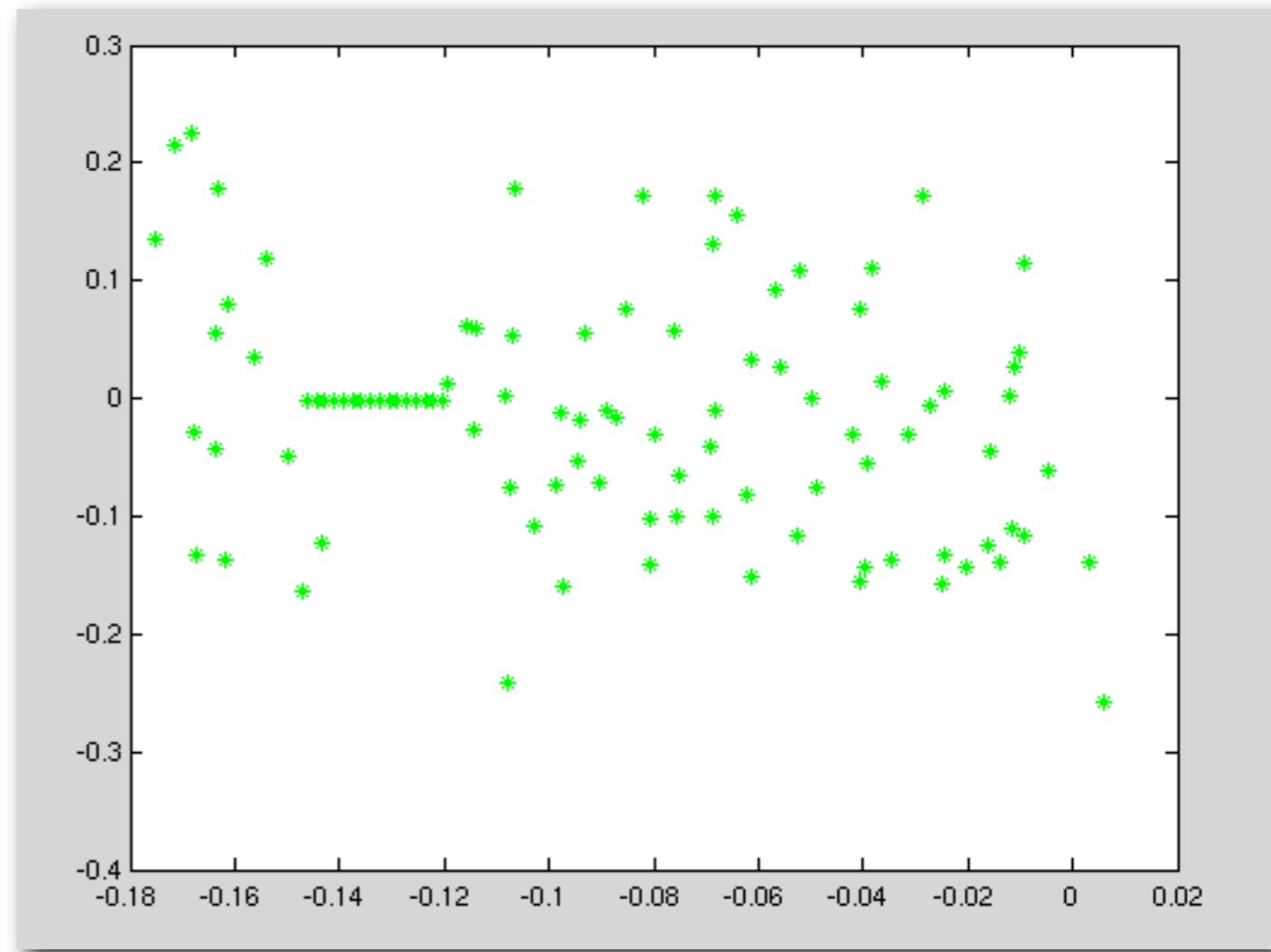
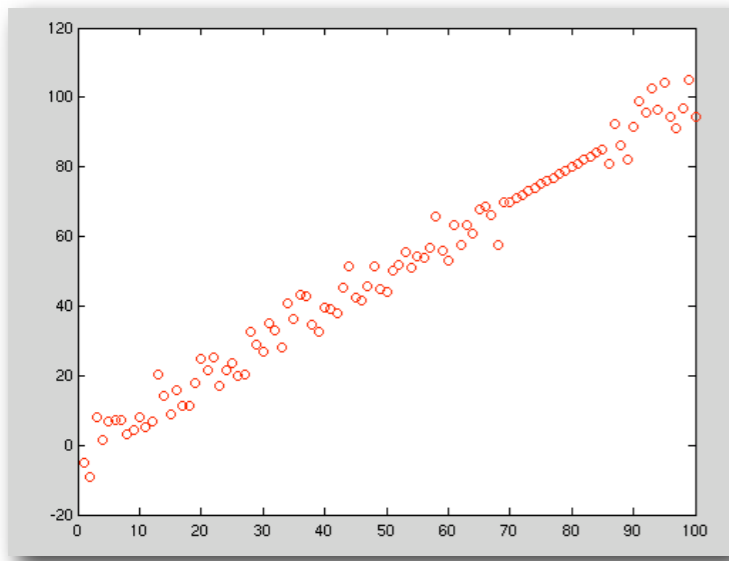
```
>> Sk=S(1:2,1:2);  
>> Uk=U(:,1:2);  
>> M = inv(Sk)*Uk';  
>> Cc = M*C;  
>> size(Uk)  
  
ans =  
  
     2     2  
  
>> size(M)  
  
ans =  
  
     2     2  
  
>> size(Cc)  
  
ans =  
  
     2    100
```



Demo

- Plot the new axes

```
>> plot(Cc(1,:),Cc(2,:), 'g*')
```



Demo

- Calculate the TFIDF score
- Plot the documents on a two term axis
- Perform SVD decomposition
 - Validate decomposition
- Reduce rank of system
- Show "M"
 - Demonstrate what SVD is capturing
- Execute a query



Demo

- Calculate the TFIDF score

	<i>wiki: baseballBat</i>	<i>wiki: bat</i>	<i>wiki: coffee</i>	<i>djp3 paper</i>	<i>starbucks</i>	<i>wiki:starbucks</i>	
tf =							df =
ball	6	0	0	0	0	0	1
baseball	10	0	0	0	0	0	1
bat	24	5	0	0	0	0	2
California	0	0	0	6	0	0	1
coffee	0	0	21	0	0	10	2
company	0	0	0	0	0	8	1
encyclopedia	1	1	1	0	0	1	4
Irvine	0	0	0	6	0	0	1
run	5	0	0	0	0	0	1
species	0	3	2	0	0	0	2
starbucks	0	0	0	0	4	14	2
stores	0	0	0	0	0	7	2
university	0	0	0	6	0	0	1
users	0	0	0	5	0	0	1



Demo

- Calculate the TFIDF score

```
>> !more computeTFIDF.m
c = 6;
tf = load('tf.txt', '-ASCII');
df = load('df.txt', '-ASCII');
tfidf = zeros(size(tf));

for i = 1:size(tf,1)
for j = 1:size(tf,2)
if tf(i,j) == 0
    tfidf(i,j) = (0) * log2(c/df(i));
else
    tfidf(i,j) = (1+log2(tf(i,j))) * log2(c/df(i));
end
end
end
```



Demo

- Calculate the TFIDF Core

	<i>wiki: baseball</i>	<i>wiki: bat</i>	<i>wiki: coffee</i>	<i>djp3 paper</i>	<i>starbucks</i>	<i>wiki:starbucks</i>
ball	9.2670	0	0	0	0	0
baseball	11.1720	0	0	0	0	0
bat	8.8520	5.2651	0	0	0	0
California	0	0	0	9.2670	0	0
coffee	0	0	8.5466	0	0	6.8501
company	0	0	0	0	0	10.3399
encyclopedia	0.5850	0.5850	0.5850	0	0	0.5850
Irvine	0	0	0	9.2670	0	0
run	8.5871	0	0	0	0	0
species	0	4.0971	3.1699	0	0	0
starbucks	0	0	0	0	4.7549	7.6195
stores	0	0	0	0	0	9.8419
university	0	0	0	9.2670	0	0
users	0	0	0	8.5871	0	0



Demo

- Perform SVD decomposition
- Validate decomposition

```
>> [U S V] = svd(tfidf);  
>> size(U)  
  
ans =  
  
    14    14  
  
>> size(S)  
  
ans =  
  
    14     6  
  
>> size(V)  
  
ans =  
  
     6     6  
  
>>
```

```
>> U*S*V'  
  
ans =  
  
    9.2670   -0.0000    0.0000    0.0000    0.0000   -0.0000  
   11.1720   -0.0000    0.0000    0.0000    0.0000    0.0000  
    8.8520    5.2651   -0.0000   -0.0000   -0.0000   -0.0000  
   -0.0000    0.0000    0.0000    9.2670   -0.0000   -0.0000  
    0.0000   -0.0000    8.5466   -0.0000    0.0000    6.8501  
    0.0000   -0.0000   -0.0000   -0.0000    0.0000   10.3399  
    0.5850    0.5850    0.5850   -0.0000    0.0000    0.5850  
   -0.0000    0.0000    0.0000    9.2670   -0.0000   -0.0000  
    8.5871   -0.0000    0.0000    0.0000    0.0000    0.0000  
   -0.0000    4.0971    3.1699    0.0000    0.0000   -0.0000  
    0.0000   -0.0000   -0.0000    0.0000    4.7549    7.6195  
    0.0000   -0.0000   -0.0000   -0.0000    0.0000    9.8419  
   -0.0000    0.0000    0.0000    9.2670   -0.0000   -0.0000  
   -0.0000    0.0000    0.0000    8.5871   -0.0000   -0.0000
```



Demo

- Reduce rank of system

```
>> Sk = S(1:3,1:3)

Sk =

    19.2339         0         0
         0    18.2035         0
         0         0    18.1004

>> Uk = U(:,1:3)

Uk =

   -0.4767   -0.0000    0.0116
   -0.5747   -0.0000    0.0140
   -0.4946    0.0000    0.0086
    0.0000   -0.5091    0.0000
   -0.0119    0.0000   -0.4755
   -0.0102    0.0000   -0.5514
   -0.0354    0.0000   -0.0383
    0.0000   -0.5091    0.0000
   -0.4417   -0.0000    0.0107
   -0.0325   -0.0000   -0.0427
   -0.0080   -0.0000   -0.4365
   -0.0097    0.0000   -0.5249
    0.0000   -0.5091    0.0000
    0.0000   -0.4717    0.0000
```



Demo

- Show "M"

```
>> M = inv(Sk)*Uk'
```

```
M =
```

```
Columns 1 through 10
```

-0.0248	-0.0299	-0.0257	0.0000	-0.0006	-0.0005	-0.0018	0.0000	-0.0230	-0.0017
-0.0000	-0.0000	0.0000	-0.0280	0.0000	0.0000	0.0000	-0.0280	-0.0000	-0.0000
0.0006	0.0008	0.0005	0.0000	-0.0263	-0.0305	-0.0021	0.0000	0.0006	-0.0024

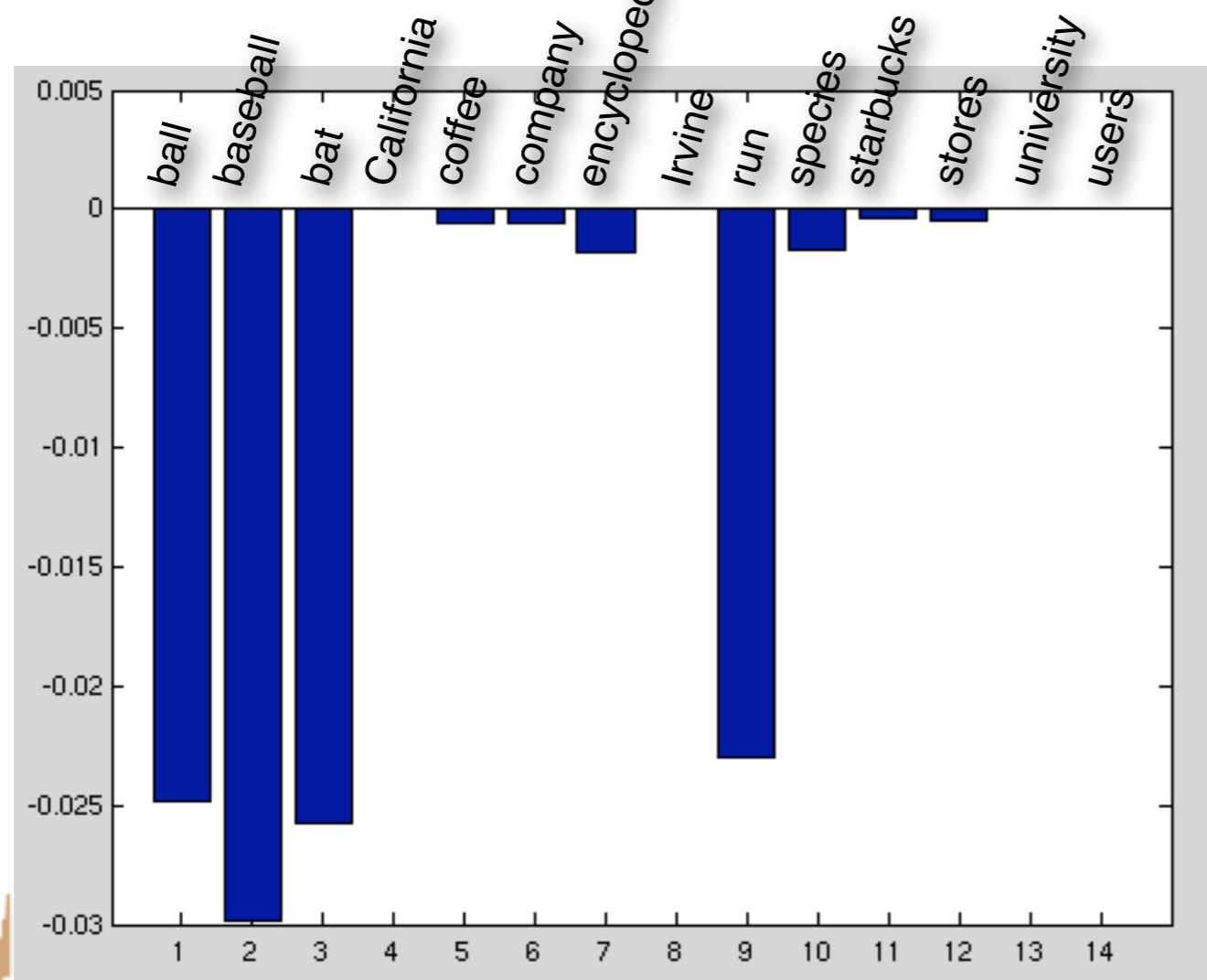
```
Columns 11 through 14
```

-0.0004	-0.0005	0.0000	0.0000
-0.0000	0.0000	-0.0280	-0.0259
-0.0241	-0.0290	0.0000	0.0000

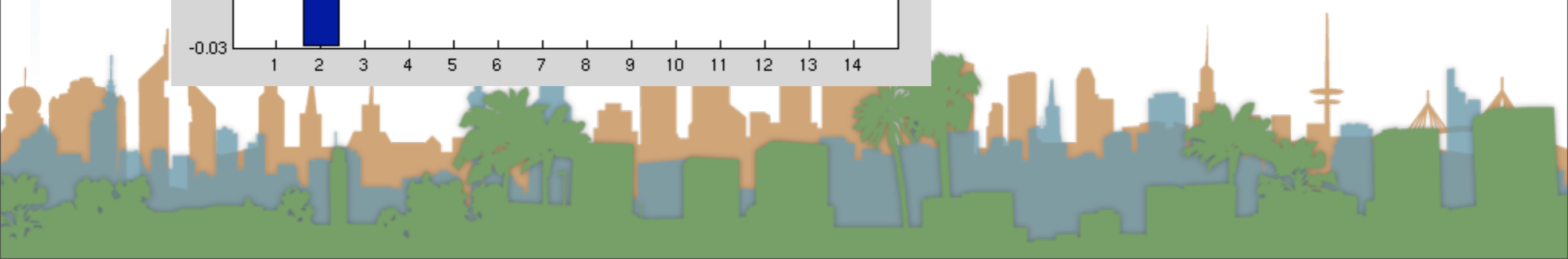


Demo

- Demonstrate what SVD is capturing
 - 1st concept (1st row of M)

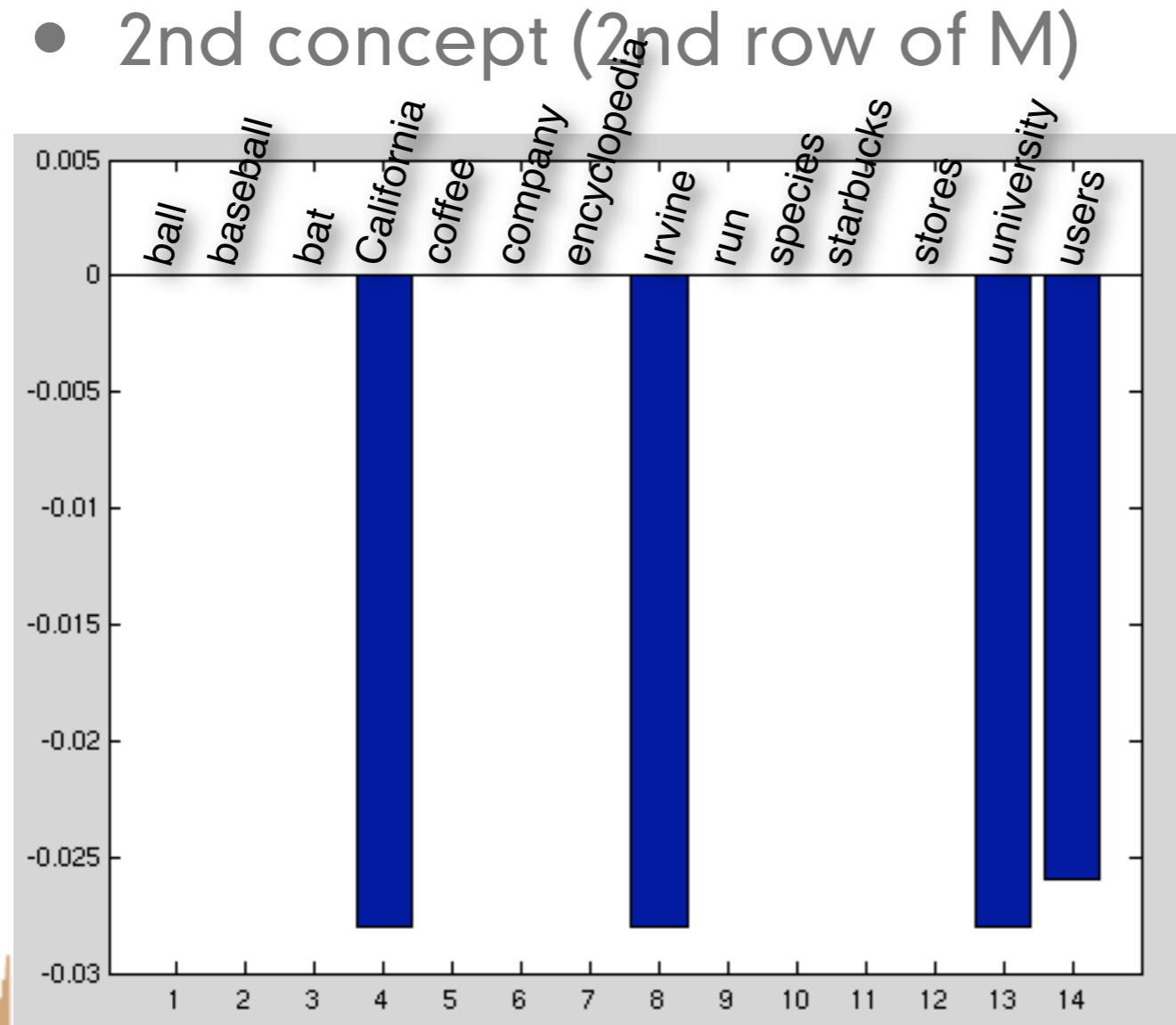


First concept is selecting for baseball?



Demo

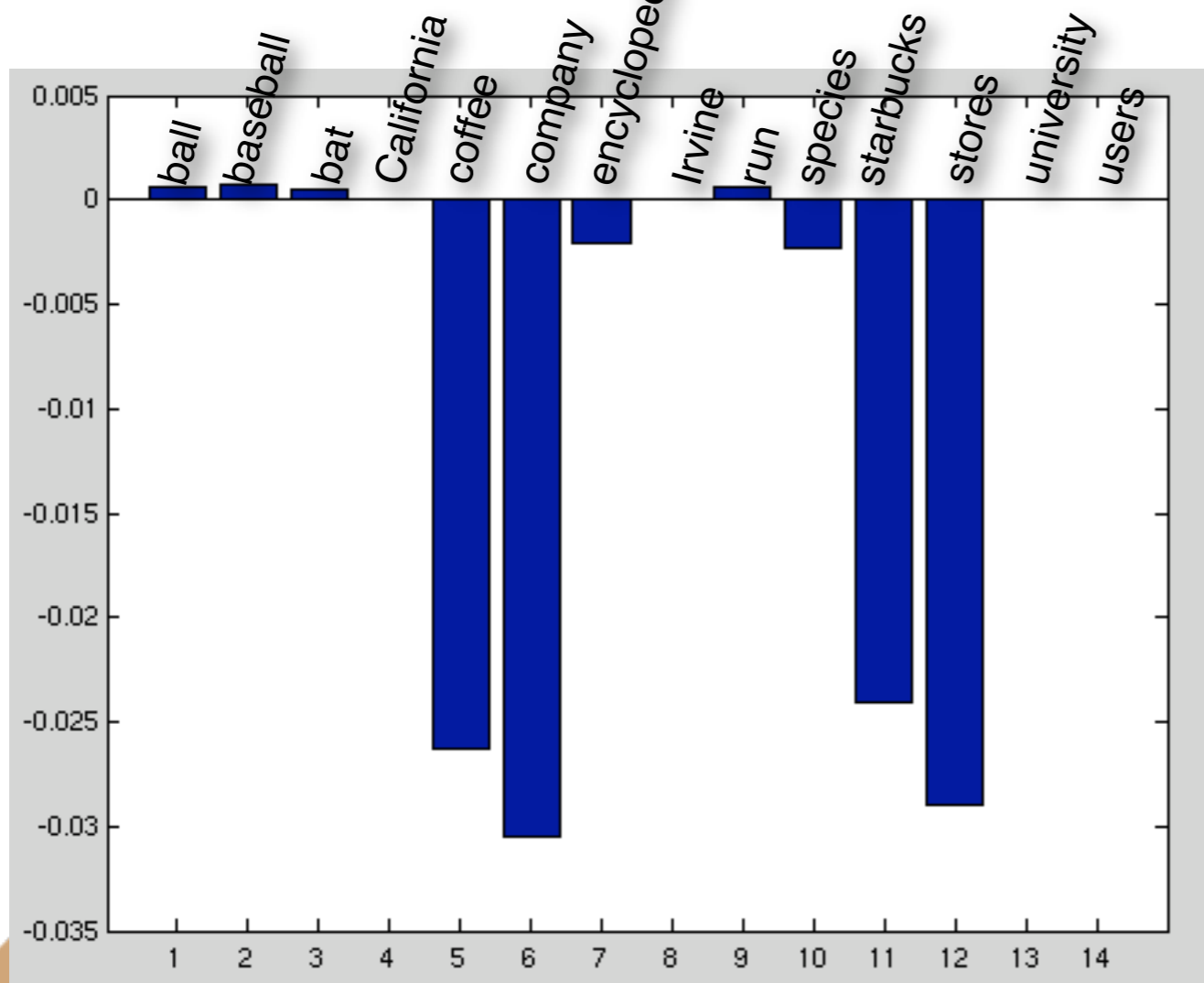
- Demonstrate what SVD is capturing
- 2nd concept (2nd row of M)



Second concept is selecting for UCI?

Demo

- Demonstrate what SVD is capturing
 - 3rd concept (3rd row of M)



Third concept is selecting for coffee?

Demo

- Execute a query "coffee stores"

```
ball baseball bat California coffee company encyclopedia Irvine run species starbucks stores university users
>> q=[ 0 0 0 0 0 (1+log2(1))*log2(c/df(5)) 0 0 0 0 0 0 (1+log(1))*log2(c/df(12)) 0 0]
q =
Columns 1 through 10
      0      0      0      0      1.5850      0      0      0      0      0
Columns 11 through 14
      0      2.5850      0      0
>> qc=M*q'
qc =
-0.0023
 0.0000
-0.1166
```



Demo

$$\text{sim}(q, d_i) = \frac{\vec{V}(q) \cdot \vec{V}(d_i)}{|\vec{V}(q)| |\vec{V}(d_i)|}$$

- Execute a query

```
>> Cc = inv(Sk)*Uk'*tfidf
Cc =
-0.9894 -0.1434 -0.0117 0.0000 -0.0020 -0.0189
0.0000 0.0000 0.0000 -1.0000 -0.0000 -0.0000
0.0227 -0.0084 -0.2332 0.0000 -0.1147 -0.9653

>> sim = qc'*Cc;
>> sim = sim ./ [norm(Cc(:,1)) norm(Cc(:,2)) norm(Cc(:,3)) norm(Cc(:,4)) norm(Cc(:,5)) norm(Cc(:,6)))]

sim =
-0.0004 0.0091 0.1166 -0.0000 0.1166 0.1166
```

wiki: baseballBat

wiki: bat

wiki: coffee

djp3 paper

starbucks

wiki: starbucks



Demo

- Execute a query “coffee stores”
 - Answer:
 - starbucks (0.1166)
 - wiki:starbucks(0.1166)
 - wiki:coffee (0.1166)
 - wiki:bat (0.0091)
 - djp3 paper (0.0)
 - wiki:baseballBat (-0.0004)



Demo

- Execute a query "baseball bat"

```
ball                                     baseball
bat California coffee company encyclopedia Irvine run species starbucks stores university users
>> q=[(1+log2(1))*log2(c/df(1)) (1+log2(1))*log2(c/df(2)) 0 0 0 0 0 0 0 0 0 0 0 0]
q =
Columns 1 through 10
    2.5850    2.5850         0         0         0         0         0         0         0         0
Columns 11 through 14
         0         0         0         0
>> qc=M*q'
qc =
   -0.1413
   -0.0000
    0.0037
```



Demo

$$\text{sim}(q, d_i) = \frac{\vec{V}(q) \cdot \vec{V}(d_i)}{|\vec{V}(q)| |\vec{V}(d_i)|}$$

- Execute a query

```
>> Cc = inv(Sk)*Uk'*tfidf
Cc =
-0.9894 -0.1434 -0.0117 0.0000 -0.0020 -0.0189
0.0000 0.0000 0.0000 -1.0000 -0.0000 -0.0000
0.0227 -0.0084 -0.2332 0.0000 -0.1147 -0.9653

>> sim = qc'*Cc;
>> sim = sim ./ [norm(Cc(:,1)) norm(Cc(:,2)) norm(Cc(:,3)) norm(Cc(:,4)) norm(Cc(:,5)) norm(Cc(:,6))]
```

sim =	0.1414	0.1408	0.0035	0.0000	-0.0012	-0.0009
-------	--------	--------	--------	--------	---------	---------

wiki: baseballBat

wiki: bat

wiki: coffee

djp3 paper

starbucks

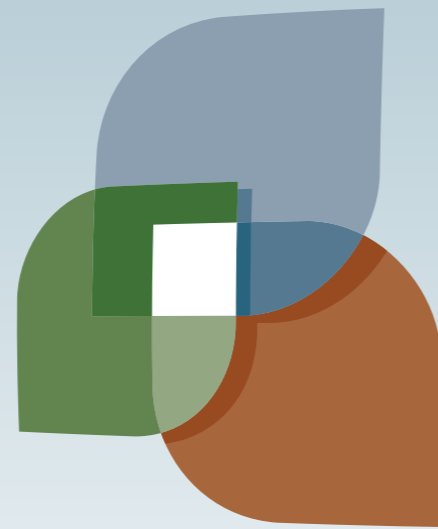
wiki: starbucks



Demo

- Execute a query “baseball bat”
 - Answer:
 - wiki:baseballBat (0.1414)
 - wiki:bat (0.1408)
 - wiki:coffee (0.0035)
 - djp3 paper (0.000)
 - wiki:starbucks (-0.0009)
 - starbucks (-0.0012)





L U C I

