Week 4

James Quacinella

09/27/2015

Exercise 3.1.3

Suppose we have a universal set U of n elements, and we choose two subsets S and T at random, each with m of the n elements. What is the expected value of the Jaccard similarity of S and T ?

Answers

The Jaccard similarity of two sets S and T is defined by the number of common elements divided by the total number of elements in the union of those sets. We can define this in terms of the number of overlapping elements, k, as such:

$$SIM(S,T) = \frac{k}{2m-k}$$

where m is the total number of elements in S or T. To find the expected value, we need to evaluate:

$$\begin{aligned} E[SIM(S,T)] &= \sum_{k=0}^{m} (SIM(S,T) * P(SIM(S,T))) \\ &= \sum_{k=0}^{m} \left(\frac{k}{2m-k} * P(SIM(S,T)) \right) \end{aligned}$$

Question is, what is the probability of having k overlapping elements? The total number of ways to choose m elements for T (assuming we have already chosen elements for S) is $\binom{n}{m}$. The total number of ways to choose elements with k elements overlapping with S is $\binom{m}{k}$ (choosing k elements from m elements in S) times $\binom{n-m}{m-k}$. Putting this all toegther we have:

$$\begin{aligned} &= \sum_{k=0}^{m} \left(\frac{k}{2m-k} * P(SIM(S,T)) \right) \\ &= \sum_{k=0}^{m} \left(\frac{k}{2m-k} * \frac{\binom{m}{k}\binom{n-m}{m-k}}{\binom{n}{m}} \right) \end{aligned}$$

I do not htink I can simplify this any more.

Exercise 3.3.3

In Fig. 3.5 is a matrix with six rows.

element	S_1	S_2	S_3	S_4
0	0	1	0	1
1	0	1	0	0
2	1	0	0	1
3	0	0	1	0
4	0	0	1	1
5	1	0	0	0

(a) Compute the minhash signature for each column if we use the following three hash functions: $h_1(x) = 2x + 1mod6; h_2(x) = 3x + 2mod6; h_3(x) = 5x + 2mod6$.

(b) Which of these hash functions are true permutations?

(c) How close are the estimated Jaccard similarities for the six pairs of columns to the true Jaccard similarities?

Answer

(a) Lets create a matrix with the has function computed, just ilke page 82 in the textbook:

row	S_1	S_2	S_3	S_4	$h_1(x)$	$h_2(x)$	$h_3(x)$
0	0	1	0	1	1	2	2
1	0	1	0	0	3	5	1
2	1	0	0	1	5	2	0
3	0	0	1	0	1	5	5
4	0	0	1	1	3	2	4
5	1	0	0	0	5	5	3

So far so good. Next we initialize a matrix that consists of all inf:

row	S_1	S_2	S_3	S_4
$h_1(x)$	∞	∞	∞	∞
$h_2(x)$	∞	∞	∞	∞
$h_3(x)$	∞	∞	∞	∞

Looking at row 0, we have 1's for S_2 and S_4 so only those columns can change. Since the values from the hash colums are all smaller than ∞, the matrix looks like this:

row	S_1	S_2	S_3	S_4
$h_1(x)$	∞	1	∞	1
$h_2(x)$	∞	2	∞	2
$h_3(x)$	∞	2	∞	2

Next row, we follow the same procedure. In this case, we have 1's for only S_2, and only the value for $h_3(x)$ changes, since its the only value less than what we have in the matrix currently:

row	S_1	S_2	S_3	S_4
$h_1(x)$	∞	1	∞	1
$h_2(x)$	∞	2	∞	2
$h_3(x)$	∞	1	∞	2

For row with label 2, we get:

row	S_1	S_2	S_3	S_4
$h_1(x)$	5	1	∞	1
$h_2(x)$	2	2	∞	2
$h_3(x)$	0	1	∞	0

For row with label 3, we get:

row	S_1	S_2	S_3	S_4
$h_1(x)$	5	1	1	1
$h_2(x)$	2	2	5	2
$h_3(x)$	0	1	5	0

For row with label 4, we get:

row	S_1	S_2	S_3	S_4
$h_1(x)$	5	1	1	1
$h_2(x)$	2	2	2	2
$h_3(x)$	0	1	4	0

For the last row, our final min-hash signature matrix looks like:

row	S_1	S_2	S_3	S_4
$h_1(x)$	5	1	1	1
$h_2(x)$	2	2	2	2
$h_3(x)$	0	1	4	0

(b) Only $h_3(x)$ is a hash function that is a true permutations since there are no collisions amoung the outputs for any of the rows. The other functions have duplicate output values.

(c) Lets calculate the real Jaccard similarities, which is the intersection carindality divided by the union cardinality:

$$\begin{array}{ll}
SIM_{real}(S_1, S_2) = \frac{0}{4} & SIM_{approx}(S_1, S_2) = \frac{1}{3} \\
SIM_{real}(S_1, S_3) = \frac{0}{4} & SIM_{approx}(S_1, S_3) = \frac{1}{3} \\
SIM_{real}(S_1, S_4) = \frac{1}{4} & SIM_{approx}(S_1, S_4) = \frac{2}{3} \\
SIM_{real}(S_2, S_3) = \frac{0}{4} & SIM_{approx}(S_2, S_3) = \frac{2}{3} \\
SIM_{real}(S_2, S_4) = \frac{1}{4} & SIM_{approx}(S_2, S_4) = \frac{2}{3} \\
SIM_{real}(S_3, S_4) = \frac{1}{4} & SIM_{approx}(S_3, S_4) = \frac{2}{3}
\end{array}$$

Exercise 3.5.5

Compute the cosines of the angles between each of the following pairs of vectors:

(a) (3, -1, 2) and (-2, 3, 1).
(b) (1, 2, 3) and (2, 4, 6).
(c) (5, 0, -4) and (-1, -6, 2).
(d) (0, 1, 1, 0, 1, 1) and (0, 0, 1, 0, 0, 0).

Answer

The cosine distance metric is the vector dot product divided by the product of the vector L_2-norms:

(a) $x = (3, -1, 2)$ and $y = (-2, 3, 1)$. The L_2 norm for x is $\sqrt{}(3^2 + (-1)^2 + 2^2) = \sqrt{}(14)$

The L_2 norm for y is $\sqrt{}((-2)^2 + 3^2 + 1^2) = \sqrt{}(14)$
The dot product $x\dot{y}$ is $(3)(-2) + (-1)(3) + (2)(1) = -6 - 3 + 2 = -7$
Therefore the cosine of the angle between x and y is $\frac{-7}{\sqrt{}(14)\sqrt{}(14)} = -0.5$.

(b) $x = (1, 2, 3)$ and $y = (2, 4, 6)$. The L_2 norm for x is $\sqrt{}(1^2 + 2^2 + 3^2) = \sqrt{}(14)$

The L_2 norm for y is $\sqrt{}(2^2 + 4^2 + 6^2) = \sqrt{}(56)$
The dot product $x\dot{y}$ is $(1)(2) + (2)(4) + (3)(6) = 28$
Therefore the cosine of the angle between x and y is $\frac{28}{\sqrt{}(14)\sqrt{}(56)} = 1$.

(c) $x = (5, 0, -4)$ and $y = (-1, -6, 2)$.. The L_2 norm for x is $\sqrt{}(5^2 + 0^2 + (-4)^2) = \sqrt{}(41)$

The L_2 norm for y is $\sqrt{}((-1)^2 + (-6)^2 + 2^2) = \sqrt{}(41)$
The dot product $x\dot{y}$ is $(5)(-1) + (0)(-6) + (-4)(2) = -13$
Therefore the cosine of the angle between x and y is $\frac{-13}{\sqrt{}(41)\sqrt{}(41)} = -0.317$.

(d) $x = (0, 1, 1, 0, 1, 1)$ and $y = (0, 0, 1, 0, 0, 0)$.. The L_2 norm for x is $\sqrt{}(4) = 2$

The L_2 norm for y is $\sqrt{}(1)$
The dot product $x\dot{y}$ is $(0)(0) + (1)(0) + (1)(1) + (0)(0) + (1)(0) + (1)(0) = 1$
Therefore the cosine of the angle between x and y is $\frac{1}{2}$.

Exercise 3.7.1

Suppose we construct the basic family of six locality-sensitive functions for vectors of length six. For each pair of the vectors 000000, 110011, 010101, and 011100, which of the six functions makes them candidates?

Answer

Lets define the 6 hash functions as $h_i(x)$, which returns the i-th bit in the vector x. Therefore, two values will collide, or make them candidates for similarity testing, when they agree on their i-th value.

So, the pairs are as follows:

000000 and 110011 will have h_3 and h_4 make them candidates

000000 and 010101 will have h_1, h_3 and h_5 make them candidates

000000 and 011100 will have h_1, h_5 and h_6 make them candidates

110011 and 010101 will have h_2, h_3 and h_6 make them candidates

110011 and 011100 will have h_2 make them candidates

010101 and 011100 will have h_1, h_2, h_4 and h_5 make them candidates

Code: Write an R function `shingle(x, k)` that generates k-shingles from a given character vector x.

```
require("sets");
```

```
## Loading required package: sets
```

```
# Terrible implementation of shingles() function
## Loops over starting positions and size k, generating substrings
## and returning a set of k-shingles
shingles <- function(x, k) {
  # Create empty set
  s <- canonicalize_set_and_mapping(c());

  # Loop over every position in string ...
  for (i in 1:nchar(x)) {
    # Loop over various shingle sizes ...
    for (j in 1:k) {
      # Then create shingles starting from position with size j
      # and do set-union with set
      s <- s + set(substring(x, i, i+j-1));
    }
  }

  # Return set
  s
}
```

```
# Slightly better but could be simpler without use of unlist, etc
shingles2 <- function(x, k) {
  canonicalize_set_and_mapping(unlist(lapply(1:nchar(x), function(l) { lapply(1:k, function(k_shin) {as
}
```

Examples:

```
print(shingles("james", 2));
```

```
## {"a" [1], "am" [1], "e" [1], "es" [1], "j" [1], "ja" [1], "m" [1],
##  "me" [1], "s" [2]}
```

```
print(shingles("This is the title of some article", 5));
```

```
## {" " [6], " a" [1], " ar" [1], " art" [1], " arti" [1], " i" [1],
##  " is" [1], " is " [1], " is t" [1], " o" [1], " of" [1], " of "
##  [1], " of s" [1], " s" [1], " so" [1], " som" [1], " some" [1], "
##  t" [2], " th" [1], " the" [1], " the " [1], " ti" [1], " tit"
##  [1], " titl" [1], "T" [1], "Th" [1], "Thi" [1], "This" [1], "This
##  " [1], "a" [1], "ar" [1], "art" [1], "arti" [1], "artic" [1], "c"
##  [1], "cl" [1], "cle" [3], "e" [8], "e " [3], "e a" [1], "e ar"
##  [1], "e art" [1], "e o" [1], "e of" [1], "e of " [1], "e t" [1],
##  "e ti" [1], "e tit" [1], "f" [1], "f " [1], "f s" [1], "f so"
##  [1], "f som" [1], "h" [2], "he" [1], "he " [1], "he t" [1], "he
##  ti" [1], "hi" [1], "his" [1], "his " [1], "his i" [1], "i" [4],
##  "ic" [1], "icl" [1], "icle" [2], "is" [2], "is " [2], "is i" [1],
##  "is is" [1], "is t" [1], "is th" [1], "it" [1], "itl" [1], "itle"
##  [1], "itle " [1], "l" [2], "le" [5], "le " [1], "le o" [1], "le
##  of" [1], "m" [1], "me" [1], "me " [1], "me a" [1], "me ar" [1],
##  "o" [2], "of" [1], "of " [1], "of s" [1], "of so" [1], "om" [1],
##  "ome" [1], "ome " [1], "ome a" [1], "r" [1], "rt" [1], "rti" [1],
##  "rtic" [1], "rticl" [1], "s" [3], "s " [2], "s i" [1], "s is"
##  [1], "s is " [1], "s t" [1], "s th" [1], "s the" [1], "so" [1],
##  "som" [1], "some" [1], "some " [1], "t" [4], "th" [1], "the" [1],
##  "the " [1], "the t" [1], "ti" [2], "tic" [1], "ticl" [1], "ticle"
##  [1], "tit" [1], "titl" [1], "title" [1], "tl" [1], "tle" [1],
##  "tle " [1], "tle o" [1]}
```

```
print(shingles2("james", 2)$set);
```

```
## {"am", "ame", "es", "ja", "jam", "me", "mes", "s"}
```

```
print(shingles2("This is the title of some article", 5)$set);
```

```
## {" a", " ar", " art", " arti", " artic", " i", " is", " is ", " is
##  t", " is th", " o", " of", " of ", " of s", " of so", " s", "
##  so", " som", " some", " some ", " t", " th", " the", " the ", "
##  the t", " ti", " tit", " titl", " title", "Th", "Thi", "This",
##  "This ", "This i", "ar", "art", "arti", "artic", "articl", "cl",
##  "cle", "e", "e ", "e a", "e ar", "e art", "e arti", "e o", "e
##  of", "e of ", "e of s", "e t", "e ti", "e tit", "e titl", "f ",
##  "f s", "f so", "f som", "f some", "he", "he ", "he t", "he ti",
##  "he tit", "hi", "his", "his ", "his i", "his is", "ic", "icl",
##  "icle", "is", "is ", "is i", "is is", "is is ", "is t", "is th",
##  "is the", "it", "itl", "itle", "itle ", "itle o", "le", "le ",
##  "le o", "le of", "le of ", "me", "me ", "me a", "me ar", "me
##  art", "of", "of ", "of s", "of so", "of som", "om", "ome", "ome
##  ", "ome a", "ome ar", "rt", "rti", "rtic", "rticl", "rticle", "s
##  ", "s i", "s is", "s is ", "s is t", "s t", "s th", "s the", "s
##  the ", "so", "som", "some", "some ", "some a", "th", "the", "the
##  ", "the t", "the ti", "ti", "tic", "ticl", "ticle", "tit",
##  "titl", "title", "title ", "tl", "tle", "tle ", "tle o", "tle
##  of"}
```