HW 02

Elizabeth Hunt

September 27, 2023

1 Question One

Computing ϵ_{mac} for single precision numbers

```
(load "../lizfcm.asd")
(ql:quickload :lizfcm)

(let ((domain-values (lizfcm.approx:compute-maceps (lambda (x) x)
                                                   1.0
                                                   1.0)))
  (lizfcm.utils:table (:headers '("a" "h" "err")
                       :domain-order (a h err)
                       :domain-values domain-values)))
```

(with many rows truncated)

a	h	err
1.0	0.5	0.5
1.0	0.25	0.25
1.0	0.125	0.125
1.0	0.0625	0.0625
1.0	0.03125	0.03125
1.0	1.9073486e-06	1.9073486e-06
1.0	9.536743e-07	9.536743e-07
1.0	4.7683716e-07	4.7683716e-07
1.0	2.3841858e-07	2.3841858e-07
1.0	1.1920929e-07	1.1920929e-07

$\epsilon_{\text{mac single precision}} \approx 1.192(10^{-7})$

2 Question Two

Computing ϵ_{mac} for double precision numbers:

```
(let ((domain-values (lizfcm.approx:compute-maceps (lambda (x) x)
                                                   1.0d0
                                                   1.0d0)))
  (lizfcm.utils:table (:headers '("a" "h" "err")
                       :domain-order (a h err)
                       :domain-values domain-values)))
```

(with many rows truncated)

a	h	err
1.0d0	0.5d0	0.5d0
1.0d0	0.25d0	0.25d0
1.0d0	0.125d0	0.125d0
1.0d0	0.0625d0	0.0625d0
1.0d0	0.03125d0	0.03125d0
1.0d0	0.015625d0	0.015625d0
1.0d0	0.0078125d0	0.0078125d0
1.0d0	0.00390625d0	0.00390625d0
1.0d0	0.001953125d0	0.001953125d0
1.0d0	7.105427357601002d-15	7.105427357601002d-15
1.0d0	3.552713678800501d-15	3.552713678800501d-15
1.0d0	1.7763568394002505d-15	1.7763568394002505d-15
1.0d0	8.881784197001252d-16	8.881784197001252d-16
1.0d0	4.440892098500626d-16	4.440892098500626d-16
1.0d0	2.220446049250313d-16	2.220446049250313d-16

Thus, $\epsilon_{\text{mac double precision}} \approx 2.220 \cdot 10^{-16}$

3 Question Three - $|\mathbf{v}|_2$

```
(let ((vs '((1 1) (2 3) (4 5) (-1 2)))
      (2-norm (lizfcm.vector:p-norm 2)))
  (lizfcm.utils:table (:headers '("x" "y" "2norm")
                       :domain-order (x y)
                       :domain-values vs)
    (funcall 2-norm (list x y))))
```

x	y	2norm
1	1	1.4142135
2	3	3.6055512
4	5	6.4031243
-1	2	2.236068

4 Question Four - $|\mathbf{v}|_1$

```
(let ((vs '((1 1) (2 3) (4 5) (-1 2)))
      (1-norm (lizfcm.vector:p-norm 1)))
  (lizfcm.utils:table (:headers '("x" "y" "1norm")
                       :domain-order (x y)
                       :domain-values vs)
    (funcall 1-norm (list x y))))
```

x	y	1norm
1	1	2
2	3	5
4	5	9
-1	2	3

5 Question Five - $|\mathbf{v}|_\infty$

```
(let ((vs '((1 1) (2 3) (4 5) (-1 2))))
  (lizfcm.utils:table (:headers '("x" "y" "max-norm")
                       :domain-order (x y)
                       :domain-values vs)
    (lizfcm.vector:max-norm (list x y))))
```

x	y	infty-norm
1	1	1
2	3	3
4	5	5
-1	2	2

6 Question Six - ||v - u|| via $|v|_2$

```
(let* ((vs '((1 1) (2 3) (4 5) (-1 2)))
       (vs2 '((7 9) (2 2) (8 -1) (4 4)))
       (2-norm (lizfcm.vector:p-norm 2)))
  (lizfcm.utils:table (:headers '("v1" "v2" "2-norm-d")
                       :domain-order (v1 v2)
                       :domain-values (mapcar (lambda (v1 v2)
                                                (list v1 v2))
                                              vs
                                              vs2))
    (lizfcm.vector:distance v1 v2 2-norm)))
```

v1	v2	2-norm
(1 1)	(7 9)	10.0
(2 3)	(2 2)	1.0
(4 5)	(8 -1)	7.2111025
(-1 2)	(4 4)	5.3851647

7 Question Seven - ||v - u|| via $|v|_1$

```
(let* ((vs '((1 1) (2 3) (4 5) (-1 2)))
       (vs2 '((7 9) (2 2) (8 -1) (4 4)))
       (1-norm (lizfcm.vector:p-norm 1)))
  (lizfcm.utils:table (:headers '("v1" "v2" "1-norm-d")
                       :domain-order (v1 v2)
                       :domain-values (mapcar (lambda (v1 v2)
                                                (list v1 v2))
                                              vs
                                              vs2))
    (lizfcm.vector:distance v1 v2 1-norm)))
```

v1	v2	1-norm-d
(1 1)	(7 9)	14
(2 3)	(2 2)	1
(4 5)	(8 -1)	10
(-1 2)	(4 4)	7

8 Question Eight - ||v - u|| via $|v|_\infty$

```
(let* ((vs '((1 1) (2 3) (4 5) (-1 2)))
     (vs2 '((7 9) (2 2) (8 -1) (4 4))))
(lizfcm.utils:table (:headers '("v1" "v2" "max-norm-d")
                     :domain-order (v1 v2)
                     :domain-values (mapcar (lambda (v1 v2)
                                              (list v1 v2))
                                            vs
                                            vs2))
  (lizfcm.vector:distance v1 v2 'lizfcm.vector:max-norm)))
```

v1	v2	max-norm-d
(1 1)	(7 9)	-6
(2 3)	(2 2)	1
(4 5)	(8 -1)	6
(-1 2)	(4 4)	-2