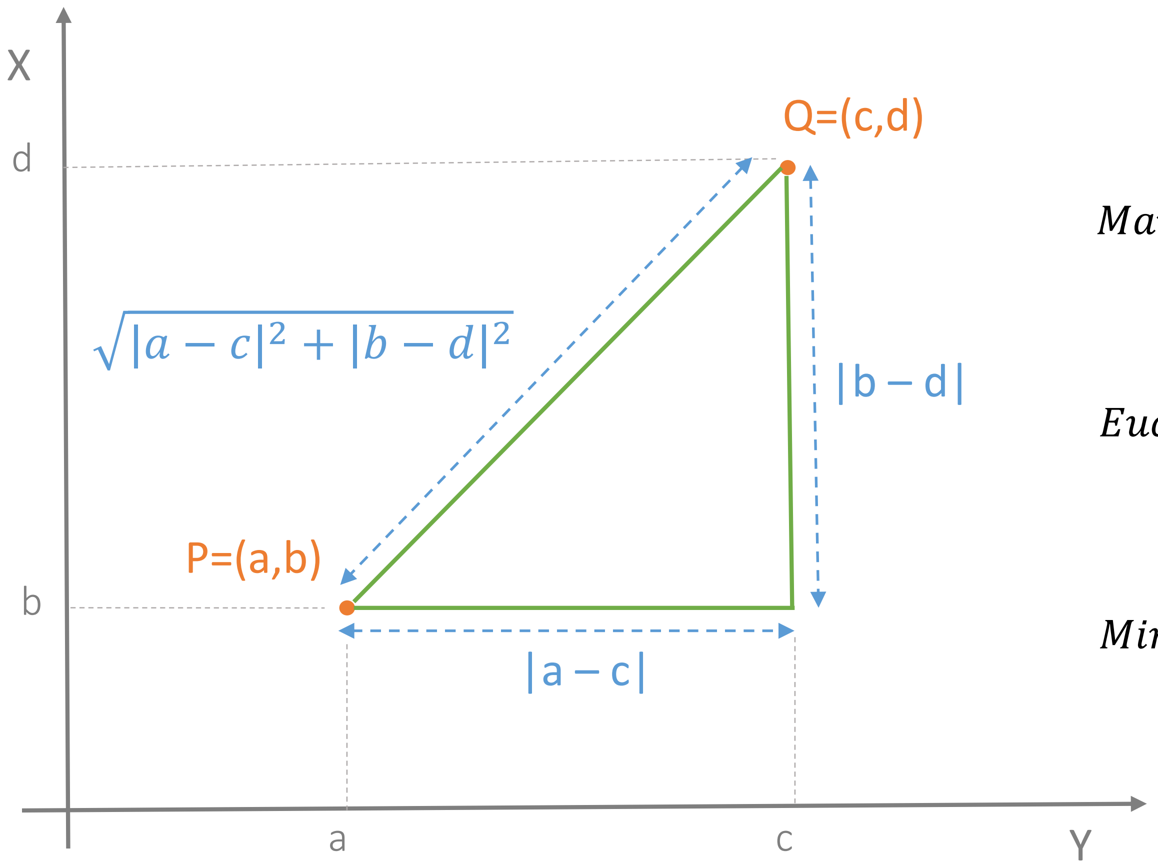
Python Class Exercise Set 2

Distance Measures

Distance Measures between Two Points P and Q: 2-Dimensional Space



$$Manhattan\ Distance = |a-c| + |b-d| = (|a-c|^1 + |b-d|^1)^{1/1}$$
$$\Rightarrow Minkowski\ Distance\ with\ r = 1$$

$$Euclidean\ Distance = \sqrt{|a-c|^2 + |b-d|^2} = (|a-c|^2 + |b-d|^2)^{1/2}$$
$$\Rightarrow Minkowski\ Distance\ with\ r = 2$$

$$Minkowski\ Distance\ (with\ parameter\ r) = (|a-c|^r + |b-d|^r)^{1/r}$$

Distance Measures between Two Points P and Q: N-Dimensional Space

$$P = (p_o, p_1, \dots, p_{n-1})$$

$$Q = (q_o, q_1, \dots, q_{n-1})$$

$$\mathit{Manhattan\ Distance} = \sum_{i=0}^{n-1} |p_i - q_i|$$

$\Rightarrow$ *Manhattan Distance with r=1*

$$\mathit{Euclidean\ Distance} = \left(\sum_{i=0}^{n-1} |p_i - q_i|^2\right)^{1/2}$$

$\Rightarrow$ *Manhattan Distance with r=2*

$$\mathit{Minkowski\ Distance} = \left(\sum_{i=0}^{n-1} |p_i - q_i|^r\right)^{1/r}$$

(with parameter r)

Note that 2-Dim Space is special case of N-Dim Space with :
$P = (a, b)$ and $Q = (c, d)$

Exercise

- Let's say we are trying to calculate the Manhattan, Euclidean, and Minkowski (r=3) Distances between two coordinates (a=1, b=2) and (c=10, d=20).

- Declare and assign four variables a, b, c, d,
  the values 1, 2, 10, 20 respectively.

- Calculate and print the Manhattan Distance between the two points:
  $|a - c| + |b - d|$ *[answer: 27]*

- Calculate and print the Euclidean Distance between the two points:
  $(|a - c|^2 + |b - d|^2)^{1/2}$ *[answer: 20.12]*

- Calculate and print the Minkowski Distance (r=3) between the two points:
  $(|a - c|^3 + |b - d|^3)^{1/3}$ *[answer: 18.72]*

Answer

```
a = 1
b = 2
c = 10
d = 20

manhattan = math.fabs(a - c) + math.fabs(b - d)
euclidean = pow(pow(math.fabs(a - c), 2) + pow(math.fabs(b - d), 2), 1/2)
minkowski = pow(pow(math.fabs(a - c), 3) + pow(math.fabs(b - d), 3), 1/3)

print ("a =", a, "b = ", b)
print ("c =", c, "d = ", d)
print ("Manhattan Distance =", round(manhattan,2))
print ("Euclidean Distance =", round(euclidean,2))
print ("Minkowski Distance =", round(minkowski,2))
```

Exercise

- Let's say we are trying to calculate the Manhattan, Euclidean, and Minkowski (r=3) Distances between two coordinates P [a=1, b=2] and Q [c=10, d=20].

- Declare two lists:

  P = [1, 2]

  Q = [10, 20]

- Update code in previous exercise to use the lists instead of variables a, b, c, and d.

Answer

```
P = [1, 2]
Q = [10, 20]

manhattan = math.fabs(P[0] - Q[0]) + math.fabs(P[1] - Q[1])
euclidean = (pow(pow(math.fabs(P[0] - Q[0]), 2) +
            pow(math.fabs(P[1] - Q[1]), 2), 1/2))
minkowski = (pow(pow(math.fabs(P[0] - Q[0]), 3) +
            pow(math.fabs(P[1] - Q[1]), 3), 1/3))

print ("P[0] =", P[0], "P[1] = ", P[1])
print ("Q[0] =", Q[0], "Q[1] = ", Q[1])
print ("Manhattan Distance =", round(manhattan,2))
print ("Euclidean Distance =", round(euclidean,2))
print ("Minkowski Distance =", round(minkowski,2))
```

Exercise

- Let's say we are trying to calculate the Manhattan, Euclidean, and Minkowski (r=3) Distances between two Lists:

  P = [1, 2, 3]

  Q = [10, 20, 30]

- Calculate and print the Manhattan, Euclidean, and Minkowski Distances between the two lists

  [Answer:

  Manhattan : 54.0

  Euclidean : 33.67

  Minkowski : 29.72]

Answer

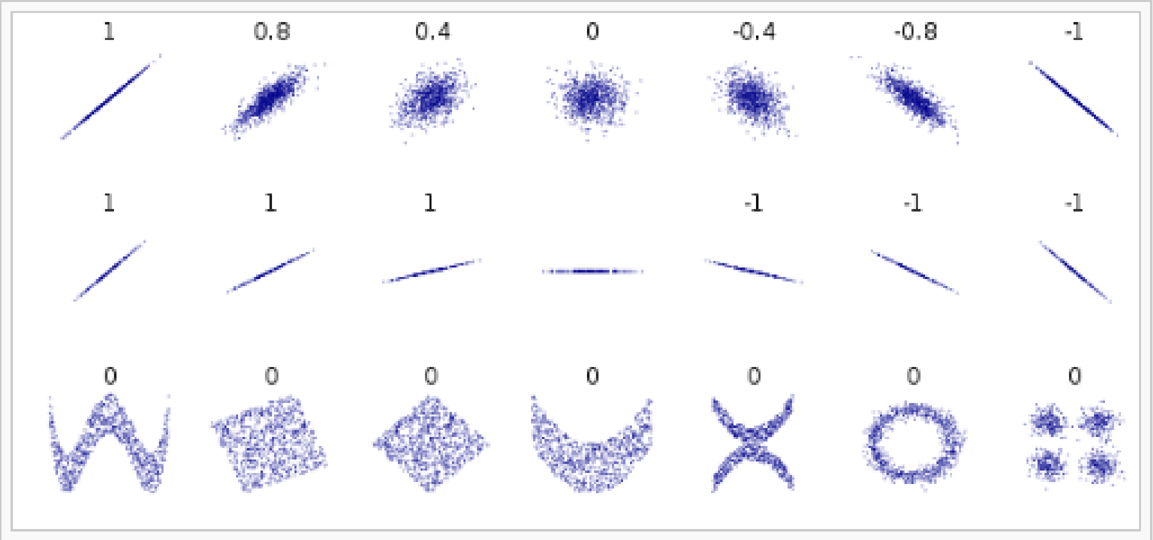
```
P = [1, 2, 3]
Q = [10, 20, 30]

manhattan = (math.fabs(P[0] - Q[0]) +
             math.fabs(P[1] - Q[1]) +
             math.fabs(P[2] - Q[2]))
euclidean = pow(pow(math.fabs(P[0] - Q[0]), 2) +
                pow(math.fabs(P[1] - Q[1]), 2) +
                pow(math.fabs(P[2] - Q[2]), 2), 1/2)
minkowski = pow(pow(math.fabs(P[0] - Q[0]), 3) +
                pow(math.fabs(P[1] - Q[1]), 3) +
                pow(math.fabs(P[2] - Q[2]), 3), 1/3)

print ("P =", P)
print ("Q =", Q)
print ("Manhattan Distance =", round(manhattan,2))
print ("Euclidean Distance =", round(euclidean,2))
print ("Minkowski Distance =", round(minkowski,2))
```

Pearson Correlation

Sample Pearson Correlation between two vectors P and Q



Measure of the linear dependence between two variables.

$$r = \frac{\sum_{i=0}^{n-1}(p_i - \bar{p})(q_i - \bar{q})}{\sqrt{\sum_{i=0}^{n-1}(p_i - \bar{p})^2}\sqrt{\sum_{i=0}^{n-1}(q_i - \bar{q})^2}}$$

Computationally efficient form:

$$r = \frac{\sum_{i=1}^{n} p_i q_i - \frac{\sum_{i=1}^{n} p_i \sum_{i=1}^{n} q_i}{n}}{\sqrt{\sum_{i=1}^{n} {p_i}^2 - \frac{\left(\sum_{i=1}^{n} p_i\right)^2}{n}}\sqrt{\sum_{i=1}^{n} {q_i}^2 - \frac{\left(\sum_{i=1}^{n} q_i\right)^2}{n}}}$$

https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient

Exercise

- Let's say we are trying to calculate the Pearson Correlation between vectors P and Q using the computationally efficient form:

  P = [1, 2, 3]

  Q = [10, 20, 30]

  n = 3

- Calculate the Partial Sums

- Calculate the Numerator and Denominator

- Calculate and print the Pearson Correlation

  [answer: 1]

Answer

```
P = [1, 2, 3]
Q = [10, 20, 30]
n = 3

sumpq = P[0]*Q[0] + P[1]*Q[1] + P[2]*Q[2]
sump = P[0] + P[1] + P[2]
sumq = Q[0] + Q[1] + Q[2]
sump2 = P[0]**2 + P[1]**2 + P[2]**2
sumq2 = Q[0]**2 + Q[1]**2 + Q[2]**2

nr = sumpq - (sump*sumq)/n
dr = pow(sump2 - (sump**2)/n, 0.5) * pow(sumq2 - (sumq**2)/n, 0.5)

r = nr/dr

print ("P =", P)
print ("Q =", Q)
print ("Pearson Correlation =", round(r,2))
```