Problem Statement:

Create an m x n matrix with replicate(m, rnorm(n)) with m=10 column vectors of n=10 elements each, constructed with rnorm(n), which creates random normal numbers.

Then we transform it into a dataframe (thus 10 observations of 10 variables) and perform an algebraic operation on each element using a nested for loop: at each iteration, every element referred by the two indexes is incremented by a sinusoidal function, compare the vectorized and non-vectorized form of creating the solution and report the system time differences.

Solution Statement:

a) m x n matrix is given by the commands and output below

```
> set.seed(2)
>
> n <- 10
> m <- 10
>
> a_matrix <- replicate(n = n,expr = rnorm(m))
> a_matrix
            [,1]        [,2]         [,3]       [,4]        [,5]       [,6]       [,7]        [,8]       [,9]       [,10]
 [1,] -0.89691455  0.41765075  2.090819205  0.7389386 -0.38358623 -0.8382871 -1.7882422 -0.92127567  0.9959846  1.600390852
 [2,]  0.18484918  0.98175278 -1.199925820  0.3189604 -1.95910318  2.0663014  2.0312425  0.33044950 -1.6957649  1.681154956
 [3,]  1.58784533 -0.39269536  1.589638200  1.0761644 -0.84170506 -0.5622471 -0.7031443 -0.14166081 -0.5333721 -1.183606388
 [4,] -1.13037567 -1.03966898  1.954651642 -0.2841577  1.90354747  1.2757155  0.1581648  0.43484776 -1.3722695 -1.358457254
 [5,] -0.08025176  1.78222896  0.004937777 -0.7766753  0.62249393 -1.0475726  0.5062348 -0.05372263 -2.2079198 -1.512670795
 [6,]  0.13242028 -2.31106908 -2.451706388 -0.5956605  1.99092044 -1.9658782 -0.8199951 -0.90711038  1.8221225 -1.253104899
 [7,]  0.70795473  0.87860458  0.477237303 -1.7259798 -0.30548372 -0.3229711 -1.9988470  1.30351223 -0.6533934  1.959357077
 [8,] -0.23969802  0.03580672 -0.596558169 -0.9025845 -0.09084424  0.9358625 -0.4792926  0.77178978 -0.2846812  0.007645872
 [9,]  1.98447394  1.01282869  0.792203270 -0.5590619 -0.18416145  1.1392298  0.0841799  1.05252560 -0.3869496 -0.842615198
[10,] -0.13878701  0.43226515  0.289636710 -0.2465126 -1.19876777  1.6716188 -0.8954866 -1.41003834  0.3866950 -0.601160105
```

As a data-frame, it becomes

```
> a_df <- data.frame(a_matrix)
> a_df
            X1          X2           X3         X4          X5         X6         X7          X8         X9         X10
1  -0.89691455  0.41765075  2.090819205  0.7389386 -0.38358623 -0.8382871 -1.7882422 -0.92127567  0.9959846  1.600390852
2   0.18484918  0.98175278 -1.199925820  0.3189604 -1.95910318  2.0663014  2.0312425  0.33044950 -1.6957649  1.681154956
3   1.58784533 -0.39269536  1.589638200  1.0761644 -0.84170506 -0.5622471 -0.7031443 -0.14166081 -0.5333721 -1.183606388
4  -1.13037567 -1.03966898  1.954651642 -0.2841577  1.90354747  1.2757155  0.1581648  0.43484776 -1.3722695 -1.358457254
5  -0.08025176  1.78222896  0.004937777 -0.7766753  0.62249393 -1.0475726  0.5062348 -0.05372263 -2.2079198 -1.512670795
6   0.13242028 -2.31106908 -2.451706388 -0.5956605  1.99092044 -1.9658782 -0.8199951 -0.90711038  1.8221225 -1.253104899
7   0.70795473  0.87860458  0.477237303 -1.7259798 -0.30548372 -0.3229711 -1.9988470  1.30351223 -0.6533934  1.959357077
8  -0.23969802  0.03580672 -0.596558169 -0.9025845 -0.09084424  0.9358625 -0.4792926  0.77178978 -0.2846812  0.007645872
9   1.98447394  1.01282869  0.792203270 -0.5590619 -0.18416145  1.1392298  0.0841799  1.05252560 -0.3869496 -0.842615198
10 -0.13878701  0.43226515  0.289636710 -0.2465126 -1.19876777  1.6716188 -0.8954866 -1.41003834  0.3866950 -0.601160105
>
```

Transformation and system time comparison is given by the commands and output below

```
> for (i in 1:m) {
+    for (j in 1:n) {
+      a_df[i,j]<-a_df[i,j] + 10*sin(0.75*pi)
+
+    }
+ }
>
> print(a_df)
         X1       X2       X3       X4       X5       X6       X7       X8       X9      X10
1  6.174153 7.488719 9.161887 7.810006 6.687482 6.232781 5.282826 6.149792 8.067052 8.671459
2  7.255917 8.052821 5.871142 7.390028 5.111965 9.137369 9.102310 7.401517 5.375303 8.752223
3  8.658913 6.678372 8.660706 8.147232 6.229363 6.508821 6.367923 6.929407 6.537696 5.887461
4  5.940692 6.031399 9.025719 6.786910 8.974615 8.346783 7.229233 7.505916 5.698798 5.712611
5  6.990816 8.853297 7.076006 6.294393 7.693562 6.023495 7.577303 7.017345 4.863148 5.558397
6  7.203488 4.759999 4.619361 6.475407 9.061988 5.105190 6.251073 6.163957 8.893190 5.817963
7  7.779023 7.949672 7.548305 5.345088 6.765584 6.748097 5.072221 8.374580 6.417674 9.030425
8  6.831370 7.106875 6.474510 6.168483 6.980224 8.006930 6.591775 7.842858 6.786387 7.078714
9  9.055542 8.083897 7.863271 6.512006 6.886906 8.210298 7.155248 8.123593 6.684118 6.228453
10 6.932281 7.503333 7.360705 6.824555 5.872300 8.742687 6.175581 5.661029 7.457763 6.469908
>
> system.time(for (i in 1:m) {
+    for (j in 1:n) {
+      a_df[i,j]<-a_df[i,j] + 10*sin(0.75*pi)
+
+    }
+ })
   user  system elapsed
   0.01    0.00    0.02
>
> system.time(a_matrix <- a_matrix + 10*sin(0.75*pi))
   user  system elapsed
      0       0       0
```