class example

1a. Compute the average distance for the flights on each airline in 2005.

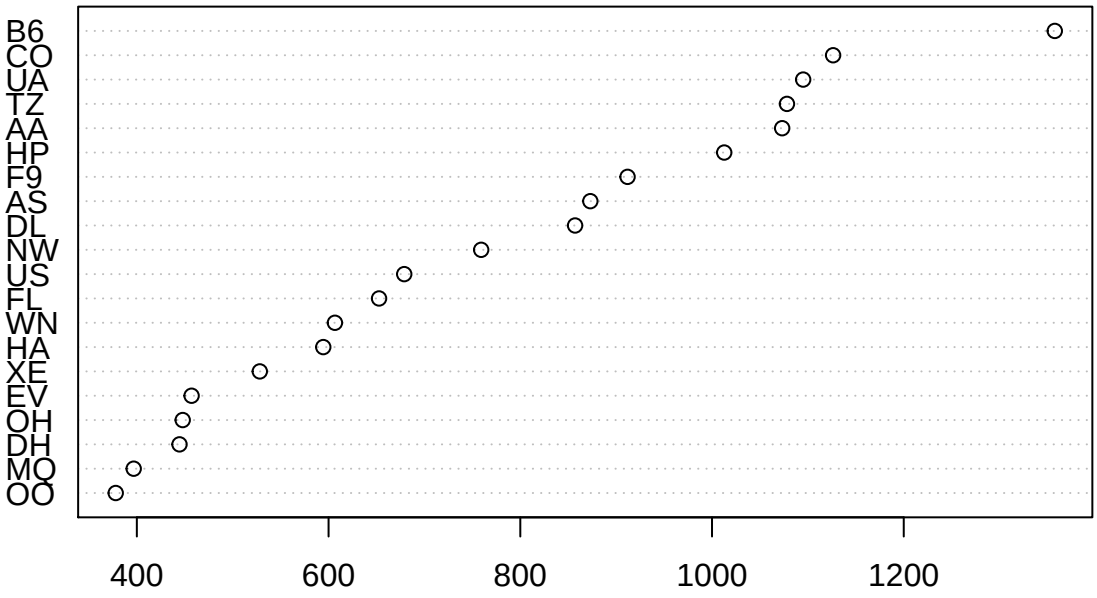
```
myDF <- read.csv("/depot/statclass/data/dataexpo2009/2005.csv")
pizza <- tapply(myDF$Distance, myDF$UniqueCarrier, mean, na.rm=T)
print(pizza)
```

```
##        AA        AS        B6        CO        DH        DL        EV
## 1073.1674  873.0624 1357.5699 1126.3239  444.5068  857.1570  457.0888
##        F9        FL        HA        HP        MQ        NW        OH
##  911.8849  652.6677  594.4586 1012.7409  396.7046  759.1927  447.8258
##        OO        TZ        UA        US        WN        XE
##  377.9543 1078.1581 1095.0739  678.9228  606.6335  528.2538
```

1b. Sort the result from 1a, and make a dotchart to display the results in sorted order. (Please display all of the values in the dotchart.)

```
dotchart(sort(pizza))
```

```
## Warning in dotchart(sort(pizza)): 'x' is neither a vector nor a matrix:
## using as.numeric(x)
```



2a. Compute the average total amount of the cost of taxi rides in June 2017, for each pickup location ID.

```
taxi <- read.csv("/depot/statclass/data/taxi2018/yellow_tripdata_2017-06.csv")
pizza2 <- tapply(taxi$total_amount,taxi$PULocationID,mean,na.rm=TRUE)
print(pizza2)
```

```
##         1         2         3         4         6         7         8
##  89.27687  50.32000  25.50333  14.81903  38.63167  13.57979  26.80241
##         9        10        11        12        13        14        15
##  17.08258  64.24437  39.87396  21.05209  20.22326  23.93232  36.90850
##        16        17        18        19        20        21        22
##  27.56415  13.95979  17.83247  24.30134  18.52806  29.38541  35.26841
##        23        24        25        26        27        28        29
```

```
##  46.97857  14.18711  16.31397  17.53992  70.74000  45.73386  45.28525
##        30        31        32        33        34        35        36
##  36.68833  24.52605  25.99459  18.91650  15.98854  19.91432  16.61088
##        37        38        39        40        41        42        43
##  15.38302  45.30088  26.85774  18.08794  13.31135  12.91456  13.84615
##        44        45        46        47        48        49        50
##  77.60750  17.01908  25.46545  12.64144  14.48305  14.82085  14.56257
##        51        52        53        54        55        56        57
##  23.19833  18.37974  21.90413  18.50768  31.36841  32.93738  23.50641
##        58        59        60        61        62        63        64
##  25.86800  29.91500  20.92972  16.29157  15.81810  31.83602  23.29958
##        65        66        67        68        69        70        71
##  18.23086  18.77628  24.48250  14.67408  16.48852  40.06867  17.01138
##        72        73        74        75        76        77        78
##  24.94337  26.58900  13.45352  13.27413  30.58784  21.42964  16.01756
##        79        80        81        82        83        84        85
##  14.33547  16.39902  18.33314  17.20068  14.92318  88.88000  16.77385
##        86        87        88        89        90        91        92
##  27.70000  21.18605  22.02639  18.63881  13.54487  23.45693  26.39135
##        93        94        95        96        97        98       100
##  58.19370  18.09831  22.90814  26.78880  15.87296  19.35000  14.51936
##       101       102       104       105       106       107       108
##  26.66867  29.51551  60.08000  25.29714  15.83596  13.78961  57.64000
##       109       111       112       113       114       115       116
##  82.80000  15.63435  16.51383  13.76005  14.47669  55.22667  15.96993
##       117       118       119       120       121       122       123
## 125.68852  55.02571  19.57504  28.51072  18.65222  25.13714  20.61568
##       124       125       126       127       128       129       130
##  97.10543  15.58651  19.28959  19.04298  19.47194  16.25566  51.95510
##       131       132       133       134       135       136       137
##  22.32296  56.16835  20.21853  32.40548  19.91764  17.33255  13.81230
##       138       139       140       141       142       143       144
##  44.51159  33.36095  14.20848  15.43736  13.57069  13.22706  15.24907
##       145       146       147       148       149       150       151
##  17.33961  15.36724  13.35716  15.94391  21.63071  31.20116  13.65784
##       152       153       154       155       156       157       158
##  14.27565  17.31011  48.82000  37.39172  87.09143  33.47228  17.06020
##       159       160       161       162       163       164       165
##  15.56849  27.04368  15.10166  15.00529  15.40006  14.96459  20.22629
##       166       167       168       169       170       171       172
##  15.10953  15.15077  14.84144  16.80006  14.64881  18.50746  43.75714
##       173       174       175       176       177       178       179
##  16.19056  23.47500  28.71519  65.80000  21.32581  11.24465  14.31995
##       180       181       182       183       184       185       186
##  38.30798  16.55931  17.34376  22.39750  73.22571  18.06784  15.08357
##       187       188       189       190       191       192       193
##  48.96667  15.95390  15.79898  17.01303  24.07857  20.27766  20.06789
##       194       195       196       197       198       200       201
##  44.31142  22.71287  24.57658  39.17137  17.57302  28.31860  36.44818
##       202       203       204       205       206       207       208
##  20.10629  50.76605  92.24000  31.63171  17.92889  15.75496  35.44853
##       209       210       211       212       213       214       215
##  19.30296  27.36667  15.24893  16.03310  24.27566  62.89600  63.04504
##       216       217       218       219       220       221       222
```

```
##  52.61447  13.46406  28.18333  66.17181  20.36078  72.75667  26.09586
##       223       224       225       226       227       228       229
##  19.71696  14.22443  15.20845  17.52442  18.85115  16.51341  13.40890
##       230       231       232       233       234       235       236
##  16.78438  16.52263  16.96433  15.56104  14.02526  16.14725  12.60905
##       237       238       239       240       241       242       243
##  12.08209  13.21961  13.27796  25.70848  17.95575  19.71307  19.52386
##       244       245       246       247       248       249       250
##  18.54191 120.35000  15.03277  18.34214  18.97629  14.10947  19.44569
##       251       252       253       254       255       256       257
##  68.32091  37.65732  37.07933  24.44255  16.58412  16.04973  20.73692
##       258       259       260       261       262       263       264
##  24.88615  18.02621  16.73681  20.51281  13.80696  13.09134  17.78484
##       265
##  86.17156
```

2b. Sort the result from 2a, and make a dotchart to display the results in sorted order. (Please ONLY display the results with value bigger than 80.)

```
dotchart(sort(pizza2[pizza2>80]))
```

```
## Warning in dotchart(sort(pizza2[pizza2 > 80])): 'x' is neither a vector nor
## a matrix: using as.numeric(x)
```