Project-4-STATC183

Nicholas Davidson

4/17/2023

```
a <- read.csv('stockData.csv', header = TRUE)

#Convert adjusted close prices into returns:
r <- (a[-1,3:ncol(a)]-a[-nrow(a),3:ncol(a)])/a[-nrow(a),3:ncol(a)]


# SPY Index data
spy <- read.csv('spy.csv', sep = ',', header = TRUE)

# Market return
Rm <- (spy[-1,3:ncol(spy)]-spy[-nrow(spy),3:ncol(spy)])/spy[-nrow(spy),3:ncol(spy)]
```

Part 1.

Compute estimates for α_i, β_i, $\sigma^2_{\epsilon_i}$, i = 1, 2, . . . , 30 by regressing each stock's return on the S&P500

```
# Initialize the Estimates
beta2 <- rep(0,30)

alpha2 <- rep(0,30)

sigma_e2 <- rep(0,30)

var_beta2 <- rep(0,30)

# Find the estimates
for(i in 1:30){
    q <- lm(r[,i] ~ Rm)
    beta2[i] <- q$coefficients[2]
    alpha2[i] <- q$coefficients[1]
    sigma_e2[i] <- summary(q)$sigma^2
    var_beta2[i] <- vcov(q)[2,2]
}

cat("The alpha_i's are", alpha2)
```

```
## The alpha_i's are 0.00712275 0.01492967 0.03153866 0.002925215 0.006067459 -0.001233695 0.0001979412 0.010
51966 0.008200688 -0.004854782 -0.004947315 0.0001714089 0.007131734 0.004237058 0.01160513 0.02601556 0.0070
15616 0.009727134 0.01503703 0.009221572 -0.01107703 -0.00648717 0.0188235 0.008017668 0.006323616 0.00816647
1 0.005567284 0.005199732 0.001626213 0.0004581792
```

```
cat("The beta_i's are", beta2)
```

```
## The beta_i's are 1.244979 1.152451 1.940076 1.151067 0.8103822 1.851836 0.7995876 0.9197325 1.138483 1.314
166 1.515356 1.420631 0.8286634 0.4058898 0.4630957 0.9989604 0.3023095 0.4960623 1.597319 0.5363579 1.002398
1.403567 0.7294139 0.7750037 1.056081 1.082093 0.4341884 0.4989621 1.025065 0.9931542
```

```
cat("The sigma_ei^2 are", sigma_e2)
```

```
## The sigma_ei^2 are 0.003946855 0.001839726 0.01130876 0.002383325 0.003247459 0.01096005 0.000779301 0.000
9254972 0.001810681 0.003264055 0.00133983 0.002428205 0.001775488 0.001097036 0.001850894 0.01223407 0.00146
2067 0.00514207 0.003592363 0.002518532 0.002980841 0.003205661 0.009683974 0.002625969 0.002188507 0.0036435
63 0.002201229 0.002325839 0.002194304 0.002525065
```

Part 2.

Construct the 30 × 30 variance covariance matrix based on the single index model.

```
# b
cov_matrix <- matrix(0, nrow = 30, ncol = 30)
sigma_m <- var(Rm)^.5
for (i in 1:30) {
  for (j in 1:30) {
    if (i == j) {
      # Variance calculation for diagonal elements
      cov_matrix[i, j] <- beta2[i]^2 * sigma_m^2 + sigma_e2[i]
    } else {
      # Covariance calculation for off-diagonal elements
      cov_matrix[i, j] <- beta2[i] * beta2[j] * sigma_m^2
    }
  }
}
cov_matrix
```

```
##               [,1]         [,2]         [,3]         [,4]         [,5]
##  [1,] 0.0058862253 0.0017952338 0.0030221588 0.0017930777 0.0012623752
##  [2,] 0.0017952338 0.0035015357 0.0027975479 0.0016598138 0.0011685538
##  [3,] 0.0030221588 0.0027975479 0.0160182531 0.0027941880 0.0019671839
##  [4,] 0.0017930777 0.0016598138 0.0027941880 0.0040411452 0.0011671503
##  [5,] 0.0012623752 0.0011685538 0.0019671839 0.0011671503 0.0040691647
##  [6,] 0.0028847022 0.0026703073 0.0044952878 0.0026671001 0.0018777106
##  [7,] 0.0012455598 0.0011529882 0.0019409802 0.0011516034 0.0008107599
##  [8,] 0.0014327160 0.0013262346 0.0022326293 0.0013246418 0.0009325836
##  [9,] 0.0017734745 0.0016416676 0.0027636400 0.0016396958 0.0011543902
## [10,] 0.0020471458 0.0018949992 0.0031901073 0.0018927232 0.0013325283
## [11,] 0.0023605499 0.0021851107 0.0036784911 0.0021824863 0.0015365294
## [12,] 0.0022129915 0.0020485190 0.0034485479 0.0020460587 0.0014404806
## [13,] 0.0012908529 0.0011949150 0.0020115612 0.0011934798 0.0008402421
## [14,] 0.0006322760 0.0005852844 0.0009852880 0.0005845815 0.0004115612
## [15,] 0.0007213886 0.0006677741 0.0011241540 0.0006669720 0.0004695664
## [16,] 0.0015561335 0.0014404796 0.0024249532 0.0014387495 0.0010129186
## [17,] 0.0004709235 0.0004359239 0.0007338494 0.0004354003 0.0003065336
## [18,] 0.0007727425 0.0007153112 0.0012041797 0.0007144521 0.0005029936
## [19,] 0.0024882289 0.0023033004 0.0038774557 0.0023005341 0.0016196382
## [20,] 0.0008355130 0.0007734166 0.0013019962 0.0007724877 0.0005438522
## [21,] 0.0015614890 0.0014454371 0.0024332989 0.0014437011 0.0010164046
## [22,] 0.0021864102 0.0020239133 0.0034071258 0.0020214825 0.0014231784
## [23,] 0.0011362466 0.0010517993 0.0017706353 0.0010505360 0.0007396058
## [24,] 0.0012072643 0.0011175388 0.0018813036 0.0011161966 0.0007858326
## [25,] 0.0016451135 0.0015228465 0.0025636126 0.0015210175 0.0010708375
## [26,] 0.0016856330 0.0015603546 0.0026267549 0.0015584805 0.0010972124
## [27,] 0.0006763582 0.0006260904 0.0010539823 0.0006253384 0.0004402552
## [28,] 0.0007772597 0.0007194927 0.0012112189 0.0007186286 0.0005059339
## [29,] 0.0015967976 0.0014781215 0.0024883210 0.0014763463 0.0010393877
## [30,] 0.0015470888 0.0014321072 0.0024108588 0.0014303871 0.0010070312
##               [,6]         [,7]         [,8]         [,9]        [,10]
##  [1,] 0.0028847022 0.0012455598 0.0014327160 0.0017734745 0.0020471458
##  [2,] 0.0026703073 0.0011529882 0.0013262346 0.0016416676 0.0018949992
##  [3,] 0.0044952878 0.0019409802 0.0022326293 0.0027636400 0.0031901073
##  [4,] 0.0026671001 0.0011516034 0.0013246418 0.0016396958 0.0018927232
##  [5,] 0.0018777106 0.0008107599 0.0009325836 0.0011543902 0.0013325283
##  [6,] 0.0152508826 0.0018526988 0.0021310828 0.0026379416 0.0030450119
##  [7,] 0.0018526988 0.0015792613 0.0009201613 0.0011390133 0.0013147785
##  [8,] 0.0021310828 0.0009201613 0.0019839207 0.0013101599 0.0015123354
##  [9,] 0.0026379416 0.0011390133 0.0013101599 0.0034324509 0.0018720307
## [10,] 0.0030450119 0.0013147785 0.0015123354 0.0018720307 0.0054249654
## [11,] 0.0035111826 0.0015160622 0.0017438637 0.0021586259 0.0024917313
## [12,] 0.0032916979 0.0014212929 0.0016348544 0.0020236898 0.0023359727
## [13,] 0.0019200696 0.0008290497 0.0009536216 0.0011804319 0.0013625886
## [14,] 0.0009404743 0.0004060790 0.0004670959 0.0005781904 0.0006674131
## [15,] 0.0010730242 0.0004633115 0.0005329281 0.0006596802 0.0007614779
## [16,] 0.0023146593 0.0009994261 0.0011495985 0.0014230201 0.0016426115
## [17,] 0.0007004717 0.0003024504 0.0003478963 0.0004306402 0.0004970939
## [18,] 0.0011494101 0.0004962935 0.0005708660 0.0007066413 0.0008156856
## [19,] 0.0037010977 0.0015980640 0.0018381869 0.0022753831 0.0026265056
## [20,] 0.0012427776 0.0005366079 0.0006172379 0.0007640423 0.0008819444
## [21,] 0.0023226254 0.0010028657 0.0011535550 0.0014279176 0.0016482647
## [22,] 0.0032521598 0.0014042211 0.0016152175 0.0019993823 0.0023079142
## [23,] 0.0016901017 0.0007297539 0.0008394058 0.0010390508 0.0011993905
## [24,] 0.0017957364 0.0007753650 0.0008918703 0.0011039936 0.0012743548
## [25,] 0.0024470120 0.0010565735 0.0012153328 0.0015043887 0.0017365364
## [26,] 0.0025072824 0.0010825971 0.0012452667 0.0015414422 0.0017793077
## [27,] 0.0010060441 0.0004343908 0.0004996618 0.0006185018 0.0007139451
## [28,] 0.0011561292 0.0004991947 0.0005742030 0.0007107720 0.0008204538
## [29,] 0.0023751449 0.0010255426 0.0011796393 0.0014602058 0.0016855355
## [30,] 0.0023012059 0.0009936172 0.0011429168 0.0014147492 0.0016330642
##              [,11]        [,12]        [,13]        [,14]        [,15]
```

```
##  [1,] 0.0023605499 0.0022129915 0.0012908529 0.0006322760 0.0007213886
##  [2,] 0.0021851107 0.0020485190 0.0011949150 0.0005852844 0.0006677741
##  [3,] 0.0036784911 0.0034485479 0.0020115612 0.0009852880 0.0011241540
##  [4,] 0.0021824863 0.0020460587 0.0011934798 0.0005845815 0.0006669720
##  [5,] 0.0015365294 0.0014404806 0.0008402421 0.0004115612 0.0004695664
##  [6,] 0.0035111826 0.0032916979 0.0019200696 0.0009404743 0.0010730242
##  [7,] 0.0015160622 0.0014212929 0.0008290497 0.0004060790 0.0004633115
##  [8,] 0.0017438637 0.0016348544 0.0009536216 0.0004670959 0.0005329281
##  [9,] 0.0021586259 0.0020236898 0.0011804319 0.0005781904 0.0006596802
## [10,] 0.0024917313 0.0023359727 0.0013625886 0.0006674131 0.0007614779
## [11,] 0.0042130280 0.0026935943 0.0015711917 0.0007695895 0.0008780550
## [12,] 0.0026935943 0.0049534224 0.0014729762 0.0007214823 0.0008231676
## [13,] 0.0015711917 0.0014729762 0.0026346846 0.0004208455 0.0004801592
## [14,] 0.0007695895 0.0007214823 0.0004208455 0.0013031710 0.0002351880
## [15,] 0.0008780550 0.0008231676 0.0004801592 0.0002351880 0.0021192289
## [16,] 0.0018940842 0.0017756846 0.0010357688 0.0005073326 0.0005788358
## [17,] 0.0005731956 0.0005373650 0.0003134486 0.0001535311 0.0001751697
## [18,] 0.0009405616 0.0008817669 0.0005143406 0.0002519305 0.0002874374
## [19,] 0.0030286059 0.0028392871 0.0016561753 0.0008112155 0.0009255478
## [20,] 0.0010169641 0.0009533935 0.0005561209 0.0002723950 0.0003107862
## [21,] 0.0019006028 0.0017817958 0.0010393335 0.0005090786 0.0005808279
## [22,] 0.0026612403 0.0024948856 0.0014552836 0.0007128162 0.0008132802
## [23,] 0.0013830091 0.0012965569 0.0007562904 0.0003704406 0.0004226503
## [24,] 0.0014694499 0.0013775943 0.0008035601 0.0003935938 0.0004490667
## [25,] 0.0020023883 0.0018772186 0.0010949943 0.0005363420 0.0006119337
## [26,] 0.0020517076 0.0019234550 0.0011219642 0.0005495522 0.0006270058
## [27,] 0.0008232452 0.0007717840 0.0004501868 0.0002205072 0.0002515853
## [28,] 0.0009460597 0.0008869214 0.0005173472 0.0002534032 0.0002891177
## [29,] 0.0019435795 0.0018220860 0.0010628350 0.0005205900 0.0005939617
## [30,] 0.0018830753 0.0017653639 0.0010297487 0.0005043839 0.0005754714
##              [,16]        [,17]        [,18]        [,19]        [,20]
##  [1,] 0.0015561335 0.0004709235 0.0007727425 0.0024882289 0.0008355130
##  [2,] 0.0014404796 0.0004359239 0.0007153112 0.0023033004 0.0007734166
##  [3,] 0.0024249532 0.0007338494 0.0012041797 0.0038774557 0.0013019962
##  [4,] 0.0014387495 0.0004354003 0.0007144521 0.0023005341 0.0007724877
##  [5,] 0.0010129186 0.0003065336 0.0005029936 0.0016196382 0.0005438522
##  [6,] 0.0023146593 0.0007004717 0.0011494101 0.0037010977 0.0012427776
##  [7,] 0.0009994261 0.0003024504 0.0004962935 0.0015980640 0.0005366079
##  [8,] 0.0011495985 0.0003478963 0.0005708660 0.0018381869 0.0006172379
##  [9,] 0.0014230201 0.0004306402 0.0007066413 0.0022753831 0.0007640423
## [10,] 0.0016426115 0.0004970939 0.0008156856 0.0026265056 0.0008819444
## [11,] 0.0018940842 0.0005731956 0.0009405616 0.0030286059 0.0010169641
## [12,] 0.0017756846 0.0005373650 0.0008817669 0.0028392871 0.0009533935
## [13,] 0.0010357688 0.0003134486 0.0005143406 0.0016561753 0.0005561209
## [14,] 0.0005073326 0.0001535311 0.0002519305 0.0008112155 0.0002723950
## [15,] 0.0005788358 0.0001751697 0.0002874374 0.0009255478 0.0003107862
## [16,] 0.0134826954 0.0003778648 0.0006200417 0.0019965326 0.0006704082
## [17,] 0.0003778648 0.0015764184 0.0001876396 0.0006041990 0.0002028817
## [18,] 0.0006200417 0.0001876396 0.0054499695 0.0009914353 0.0003329103
## [19,] 0.0019965326 0.0006041990 0.0009914353 0.0067847817 0.0010719704
## [20,] 0.0006704082 0.0002028817 0.0003329103 0.0010719704 0.0028784849
## [21,] 0.0012529249 0.0003791653 0.0006221756 0.0020034039 0.0006727154
## [22,] 0.0017543561 0.0005309105 0.0008711756 0.0028051831 0.0009419419
## [23,] 0.0009117141 0.0002759067 0.0004527377 0.0014578142 0.0004895139
## [24,] 0.0009686981 0.0002931514 0.0004810347 0.0015489305 0.0005201095
## [25,] 0.0013200244 0.0003994712 0.0006554958 0.0021106948 0.0007087422
## [26,] 0.0013525369 0.0004093103 0.0006716408 0.0021626817 0.0007261987
## [27,] 0.0005427038 0.0001642353 0.0002694951 0.0008677734 0.0002913864
## [28,] 0.0006236662 0.0001887364 0.0003096993 0.0009972308 0.0003348564
## [29,] 0.0012812562 0.0003877390 0.0006362443 0.0020487051 0.0006879270
## [30,] 0.0012413703 0.0003756686 0.0006164378 0.0019849283 0.0006665116
##              [,21]        [,22]        [,23]        [,24]        [,25]
##  [1,] 0.0015614890 0.0021864102 0.0011362466 0.0012072643 0.0016451135
##  [2,] 0.0014454371 0.0020239133 0.0010517993 0.0011175388 0.0015228465
```

```
##  [3,] 0.0024332989 0.0034071258 0.0017706353 0.0018813036 0.0025636126
##  [4,] 0.0014437011 0.0020214825 0.0010505360 0.0011161966 0.0015210175
##  [5,] 0.0010164046 0.0014231784 0.0007396058 0.0007858326 0.0010708375
##  [6,] 0.0023226254 0.0032521598 0.0016901017 0.0017957364 0.0024470120
##  [7,] 0.0010028657 0.0014042211 0.0007297539 0.0007753650 0.0010565735
##  [8,] 0.0011535550 0.0016152175 0.0008394058 0.0008918703 0.0012153328
##  [9,] 0.0014279176 0.0019993823 0.0010390508 0.0011039936 0.0015043887
## [10,] 0.0016482647 0.0023079142 0.0011993905 0.0012743548 0.0017365364
## [11,] 0.0019006028 0.0026612403 0.0013830091 0.0014694499 0.0020023883
## [12,] 0.0017817958 0.0024948856 0.0012965569 0.0013775943 0.0018772186
## [13,] 0.0010393335 0.0014552836 0.0007562904 0.0008035601 0.0010949943
## [14,] 0.0005090786 0.0007128162 0.0003704406 0.0003935938 0.0005363420
## [15,] 0.0005808279 0.0008132802 0.0004226503 0.0004490667 0.0006119337
## [16,] 0.0012529249 0.0017543561 0.0009117141 0.0009686981 0.0013200244
## [17,] 0.0003791653 0.0005309105 0.0002759067 0.0002931514 0.0003994712
## [18,] 0.0006221756 0.0008711756 0.0004527377 0.0004810347 0.0006554958
## [19,] 0.0020034039 0.0028051831 0.0014578142 0.0015489305 0.0021106948
## [20,] 0.0006727154 0.0009419419 0.0004895139 0.0005201095 0.0007087422
## [21,] 0.0042380776 0.0017603938 0.0009148519 0.0009720320 0.0013245674
## [22,] 0.0017603938 0.0056705795 0.0012809834 0.0013610474 0.0018546705
## [23,] 0.0009148519 0.0012809834 0.0103496832 0.0007073172 0.0009638462
## [24,] 0.0009720320 0.0013610474 0.0007073172 0.0033774950 0.0010240885
## [25,] 0.0013245674 0.0018546705 0.0009638462 0.0010240885 0.0035840106
## [26,] 0.0013571918 0.0019003515 0.0009875859 0.0010493120 0.0014298753
## [27,] 0.0005445716 0.0007625137 0.0003962677 0.0004210352 0.0005737357
## [28,] 0.0006258126 0.0008762681 0.0004553842 0.0004838466 0.0006593276
## [29,] 0.0012856657 0.0018002001 0.0009355386 0.0009940117 0.0013545187
## [30,] 0.0012456426 0.0017441593 0.0009064150 0.0009630678 0.0013123521
##              [,26]        [,27]        [,28]        [,29]        [,30]
##  [1,] 0.0016856330 0.0006763582 0.0007772597 0.0015967976 0.0015470888
##  [2,] 0.0015603546 0.0006260904 0.0007194927 0.0014781215 0.0014321072
##  [3,] 0.0026267549 0.0010539823 0.0012112189 0.0024883210 0.0024108588
##  [4,] 0.0015584805 0.0006253384 0.0007186286 0.0014763463 0.0014303871
##  [5,] 0.0010972124 0.0004402552 0.0005059339 0.0010393877 0.0010070312
##  [6,] 0.0025072824 0.0010060441 0.0011561292 0.0023751449 0.0023012059
##  [7,] 0.0010825971 0.0004343908 0.0004991947 0.0010255426 0.0009936172
##  [8,] 0.0012452667 0.0004996618 0.0005742030 0.0011796393 0.0011429168
##  [9,] 0.0015414422 0.0006185018 0.0007107720 0.0014602058 0.0014147492
## [10,] 0.0017793077 0.0007139451 0.0008204538 0.0016855355 0.0016330642
## [11,] 0.0020517076 0.0008232452 0.0009460597 0.0019435795 0.0018830753
## [12,] 0.0019234550 0.0007717840 0.0008869214 0.0018220860 0.0017653639
## [13,] 0.0011219642 0.0004501868 0.0005173472 0.0010628350 0.0010297487
## [14,] 0.0005495522 0.0002205072 0.0002534032 0.0005205900 0.0005043839
## [15,] 0.0006270058 0.0002515853 0.0002891177 0.0005939617 0.0005754714
## [16,] 0.0013525369 0.0005427038 0.0006236662 0.0012812562 0.0012413703
## [17,] 0.0004093103 0.0001642353 0.0001887364 0.0003877390 0.0003756686
## [18,] 0.0006716408 0.0002694951 0.0003096993 0.0006362443 0.0006164378
## [19,] 0.0021626817 0.0008677734 0.0009972308 0.0020487051 0.0019849283
## [20,] 0.0007261987 0.0002913864 0.0003348564 0.0006879270 0.0006665116
## [21,] 0.0013571918 0.0005445716 0.0006258126 0.0012856657 0.0012456426
## [22,] 0.0019003515 0.0007625137 0.0008762681 0.0018002001 0.0017441593
## [23,] 0.0009875859 0.0003962677 0.0004553842 0.0009355386 0.0009064150
## [24,] 0.0010493120 0.0004210352 0.0004838466 0.0009940117 0.0009630678
## [25,] 0.0014298753 0.0005737357 0.0006593276 0.0013545187 0.0013123521
## [26,] 0.0051086567 0.0005878670 0.0006755670 0.0013878808 0.0013446756
## [27,] 0.0005878670 0.0024371095 0.0002710704 0.0005568855 0.0005395495
## [28,] 0.0006755670 0.0002710704 0.0026373488 0.0006399636 0.0006200413
## [29,] 0.0013878808 0.0005568855 0.0006399636 0.0035090411 0.0012738092
## [30,] 0.0013446756 0.0005395495 0.0006200413 0.0012738092 0.0037592200
```

Part 3.

Answer the same question as in project 2, part (e) using the new inputs from (1) above. Draw the frontier on the same plot as in project 2. Now you will have two frontiers, one using the historical variance covariance matrix (project 2) and one using the variance covariance matrix with inputs from the single index model.

```
# From Project 2 part (e)
#Compute mean vector:
means <- colMeans(r) #Without ^GSPC
#Compute variance covariance matrix:
covmat <- cov(r) #Without ^GSPC
#Compute correlation matrix:
cormat <- cor(r) #Without ^GSPC
#Compute the vector of variances:
variances <- diag(covmat)
variances2 <- diag(cov_matrix)

#Compute the vector of standard deviations:
stdev <- diag(covmat)^.5
stdev2 <- diag(cov_matrix)^.5
ones <- rep(1, ncol(r))
A <- t(ones) %*% solve(covmat) %*% means
A2 <- t(ones) %*% solve(cov_matrix) %*% means
B <- t(means) %*% solve(covmat) %*% means
B2 <- t(means) %*% solve(cov_matrix) %*% means
C <- t(ones) %*% solve(covmat) %*% ones
C2 <- t(ones) %*% solve(cov_matrix) %*% ones
D <- B*C-A^2
D2 <- B2 * C2-A2^2
#Hyperbola:
#Efficient frontier:
minvar <- 1/C
minvar2 <- 1/C2
minE <- A/C
minE <- A2/C2
sdeff <- seq((minvar)^0.5, 1, by = 0.0001)
```

```
## Warning in from + (0L:n) * by: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
sdeff2 <- seq((minvar2)^0.5, 1, by = 0.0001)
```

```
## Warning in from + (0L:n) * by: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
# options(warn = -1)
y1 <- (A + sqrt(D*(C*sdeff^2 - 1)))*(1/C)
```

```
## Warning in C * sdeff^2: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in D * (C * sdeff^2 - 1): Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in sqrt(D * (C * sdeff^2 - 1)): NaNs produced
```

```
## Warning in A + sqrt(D * (C * sdeff^2 - 1)): Recycling array of length 1 in array-vector arithmetic is depr
ecated.
##   Use c() or as.vector() instead.
```

```
## Warning in (A + sqrt(D * (C * sdeff^2 - 1))) * (1/C): Recycling array of length 1 in vector-array arithmet
ic is deprecated.
##   Use c() or as.vector() instead.
```

```
y2 <- (A - sqrt(D*(C*sdeff^2 - 1)))*(1/C)
```

```
## Warning in C * sdeff^2: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in D * (C * sdeff^2 - 1): Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in sqrt(D * (C * sdeff^2 - 1)): NaNs produced
```

```
## Warning in A - sqrt(D * (C * sdeff^2 - 1)): Recycling array of length 1 in array-vector arithmetic is depr
ecated.
##   Use c() or as.vector() instead.
```

```
## Warning in (A - sqrt(D * (C * sdeff^2 - 1))) * (1/C): Recycling array of length 1 in vector-array arithmet
ic is deprecated.
##   Use c() or as.vector() instead.
```

```
y3 <- (A2 + sqrt(D2*(C2*sdeff2^2 - 1)))*(1/C2)
```

```
## Warning in C2 * sdeff2^2: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in D2 * (C2 * sdeff2^2 - 1): Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in sqrt(D2 * (C2 * sdeff2^2 - 1)): NaNs produced
```

```
## Warning in A2 + sqrt(D2 * (C2 * sdeff2^2 - 1)): Recycling array of length 1 in array-vector arithmetic is
deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in (A2 + sqrt(D2 * (C2 * sdeff2^2 - 1))) * (1/C2): Recycling array of length 1 in vector-array ari
thmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
y4 <- (A2 - sqrt(D2*(C2*sdeff2^2 - 1)))*(1/C2)
```

```
## Warning in C2 * sdeff2^2: Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in D2 * (C2 * sdeff2^2 - 1): Recycling array of length 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in sqrt(D2 * (C2 * sdeff2^2 - 1)): NaNs produced
```

```
## Warning in A2 - sqrt(D2 * (C2 * sdeff2^2 - 1)): Recycling array of length 1 in array-vector arithmetic is
deprecated.
##   Use c() or as.vector() instead.
```

```
## Warning in (A2 - sqrt(D2 * (C2 * sdeff2^2 - 1))) * (1/C2): Recycling array of length 1 in vector-array ari
thmetic is deprecated.
##   Use c() or as.vector() instead.
```

```
# options(warn = 0)
plot(sdeff, y1, type = "n",xlim=c(0,0.3), ylim=c(-0.05,0.10), xlab="Portfolio standard deviation", ylab="Expe
cted return", main = "Comparing Single Index Model to Historical Variance Covaraince", xaxt="no", yaxt="no")
axis(1, at=seq(0, 0.3, 0.02))
axis(2, at=seq(-0.05,0.10, 0.02))
points(sdeff, y1, lwd=5,type = "l", col= 'red')
points(sdeff, y2, lwd=5,type = "l", col = 'red')
points(sdeff2, y3, lwd=5,type = "l", col = 'blue')
points(sdeff2, y4, lwd=5,type = "l", col = 'blue')
legend("topright", legend=c("Project 2 part (e)", "Using Single Index Model"), col=c("red", "blue"), lty=c(1,
1), lwd=c(2,2))
```