Eigenvectors of Symmetric Matrices

Ali Taqi

11/4/2020

Computational Evidence: Real Symmetric Matrices have Real Eigenvectors

In this document, we hope to show that given any arbitrary element of the set of $M \times M$ symmetric matrices, denote it $S \in \mathcal{SM}_{\mathbb{R}}[M \times M]$ has a set of real eigenvectors $[\lambda_i] \in \mathbb{R}^M$.

To simulate a generic element S, we use the following method:

(1) First, pick some $f \in [0, 1]$, letting it denote the fraction of positive entries of S. That is;

$$\text{Want: } f \approx \frac{|\{s_{ij} > 0\}|}{M^2}$$

We hope to show that our condition is invariant to the value of f, since there is the possibility that the sign proportions of our matrix S influences the $\det(S)$.

(2) To simulate a symmetric matrix S with a fraction of positive entries f, we will sample from the distribution:

$$s_{ij} \sim \text{Unif}(f - 1, f)$$

(3) To not constrict the sizes of $|s_{ij}|$, we will add an ϵ term and scale our endpoints to preserve the fraction f.

$$s_{ij} \sim \text{Unif}(\epsilon(f - 1),\, \epsilon f)$$

(4) Having our uniform distribution, we will generate M^2 entries and insert them in the matrix S. Then, we delete the lower triangular matrix, then duplicate the entries from the upper triangle to the lower triangle.

(5) Now, if we let $f \sim \text{Unif}(0, 1)$ and let $\epsilon \to \infty$, we can well approximate $S \in \mathcal{SM}_{\mathbb{R}}[M \times M]$.

```r
RM_symm(5,0.5,10)
```

```
##            [,1]      [,2]      [,3]       [,4]       [,5]
## [1,] -0.6251419 -4.080202  1.421978  0.3695851 -3.2072968
## [2,] -4.0802022 -2.704363  3.526449 -3.2612270 -2.1405177
## [3,]  1.4219784 -3.261227 -4.310289  2.1499256 -4.0540433
## [4,]  3.5264488  2.149926 -2.140518  0.1084924  0.6325161
## [5,]  0.3695851 -3.207297 -4.054043  0.6325161  0.7438932
```

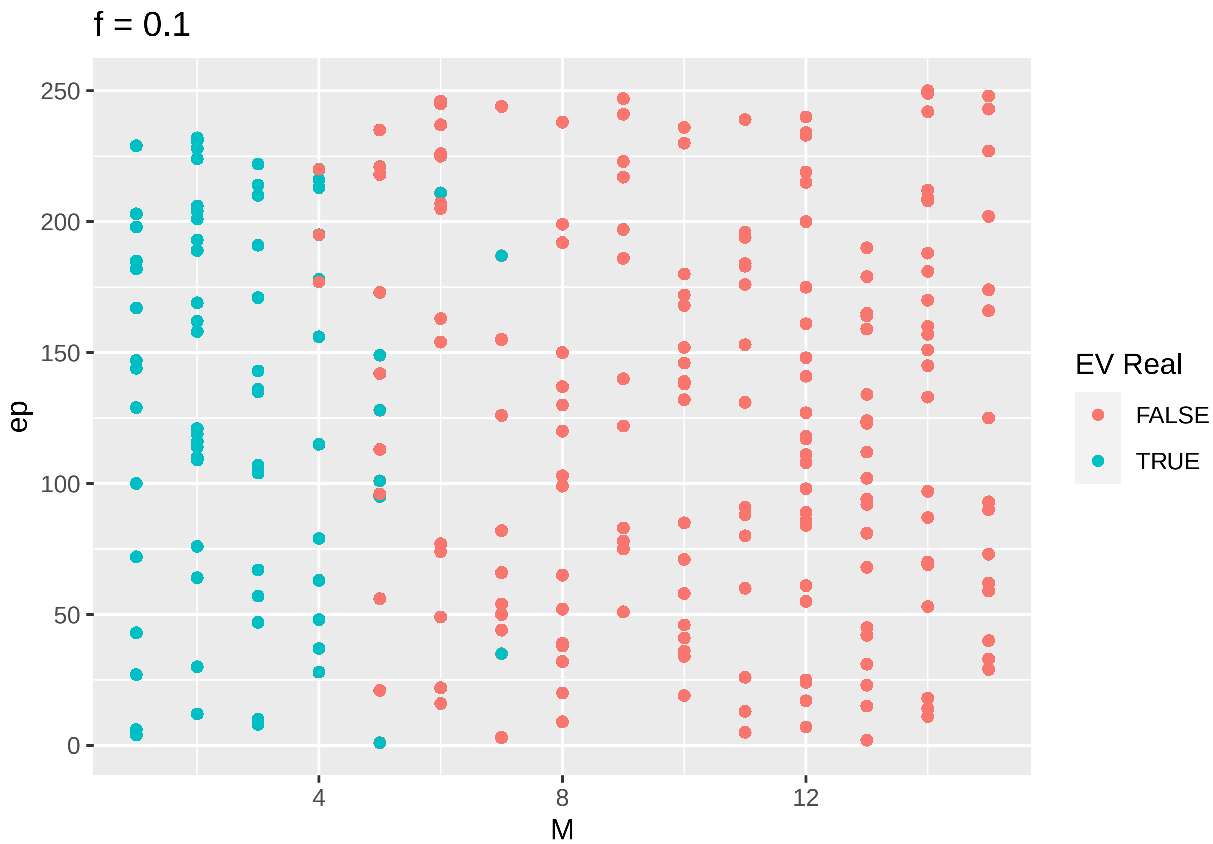
Simulation

```r
simulate_by_f <- function(f,M_max,ep_max,draws){
  M_vec <- sample(1:M_max, draws, replace = T)
  ep_vec <- sample(1:ep_max, draws, replace = F)
  table <- data.frame(M = M_vec, ep = rep(ep_vec,length(M_vec)))

  bool_vec <- rep(NA, length(table$M))

  for(i in 1:length(table$M)){
    S_curr <- RM_symm(table$M[i],f,table$ep[i])
    bool_vec[i] <- check_real_eigenvectors(eigen_frame(S_curr))
  }
  cbind(table,bool_vec)
}

plot_f_table <- function(table, f){
  ggplot() +
    geom_point(data = table, aes(x=M, y=ep, color = factor(bool_vec))) +
    labs(color = "EV Real", title = paste("f = ",f,sep=""))
}
```

```r
table <- simulate_by_f(f = 0.1, M_max = 15, ep_max = 250, draws = 250)
head(table)
```

```
##    M  ep bool_vec
## 1  1   6     TRUE
## 2  3 104     TRUE
## 3 12 118    FALSE
## 4  8  52    FALSE
## 5 13  81    FALSE
## 6  9  78    FALSE
```

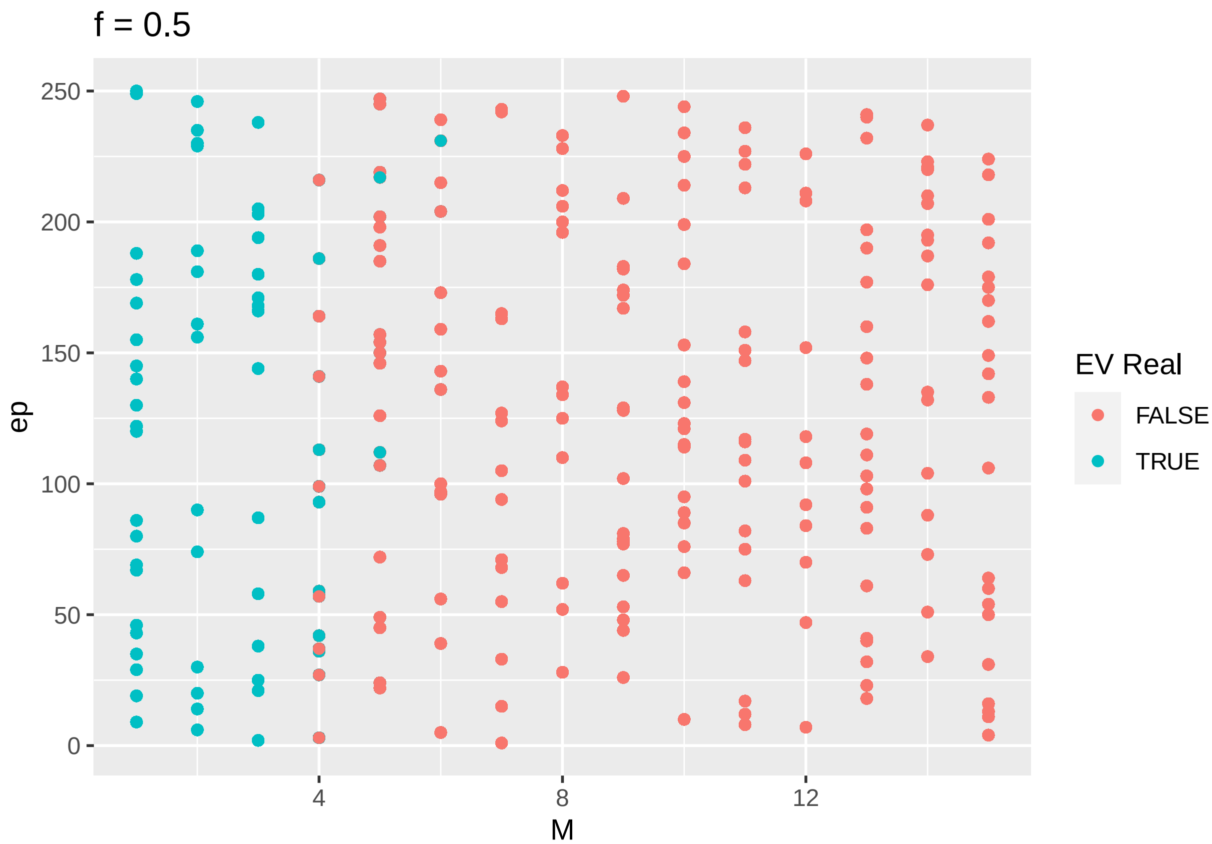
```r
plot_f_table(table, f = 0.1)
```




```
}
table <- simulate_by_f(f = 0.5, M_max = 15, ep_max = 250, draws = 250)
head(table)
```

```
##    M  ep bool_vec
## 1  3 144     TRUE
## 2  8 125    FALSE
## 3 13  18    FALSE
## 4 14 223    FALSE
## 5  6 143    FALSE
## 6 15 175    FALSE
```

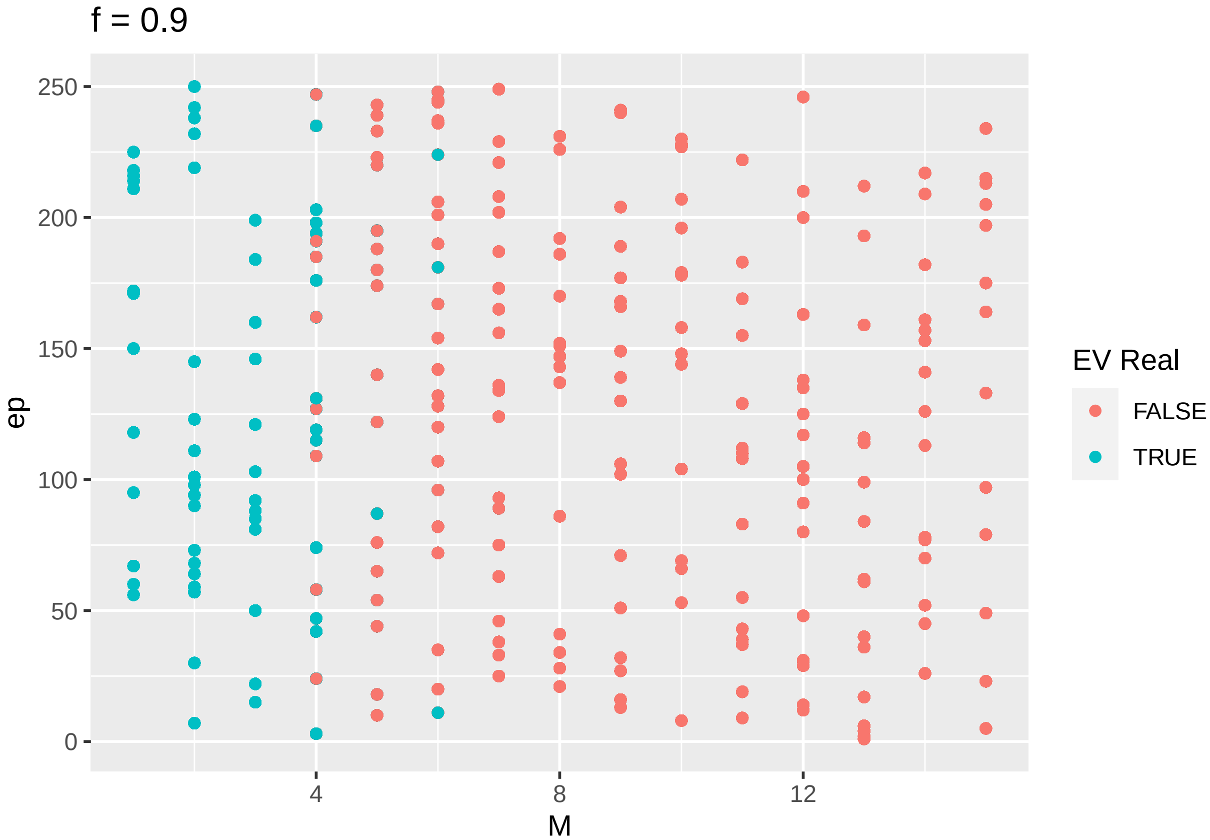
```
plot_f_table(table, f = 0.5)
```




```r
table <- simulate_by_f(f = 0.9, M_max = 15, ep_max = 250, draws = 250)
head(table)
```

```
##    M  ep bool_vec
## 1  2 219     TRUE
## 2  1 211     TRUE
## 3  8 147    FALSE
## 4 10 227    FALSE
## 5  5  18    FALSE
## 6  6 154     TRUE
```

```r
plot_f_table(table, f = 0.9)
```


f = 0.9
ep
250
200
150
100
50
0
4
8
12
M
EV Real
FALSE
TRUE

“‘