Eigenvectors of Symmetric Matrices

Ali Taqi

11/4/2020

```
S <- RM_symm(8,0.5,10)
prop <- prop_real_rows(eigen_frame = eigen_frame(S))
prop
```

```
## # A tibble: 8 x 3
##   row_i prop_reals is_real
##   <dbl>      <dbl> <lgl>
## 1     1       0.25 FALSE
## 2     2       0.25 FALSE
## 3     3       0.25 FALSE
## 4     4       0.5  FALSE
## 5     5       0.25 FALSE
## 6     6       0.5  FALSE
## 7     7       0.5  FALSE
## 8     8       0.25 FALSE
```

```
avgprop_real_components(eigen_frame(S))
```

```
## [1] 0.34375
```

Simulation

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

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

  for(i in 1:length(table$M)){
    S_curr <- RM_symm(table$M[i],f,ep)
    prop <- avgprop_real_components(eigen_frame(S_curr))
    #print(prop)
    prop_vec[i] <- prop
  }
  cbind(table,prop_vec)
}

plot_f_table <- function(table, f){
  ggplot() +
    geom_point(data = table, aes(x=M, y=prop_vec, color = prop_vec)) +
    labs(color = "EV Real", title = paste("f = ",f,sep="")) +
    scale_color_gradient(high="blue", low="red")
```

```
}
```

```
M <- 200
ep <- 100
d <- ep
```

```
f <- 0.5
table <- simulate_by_f(f = f, M_max = M, ep = ep, draws = d)
head(table)
```

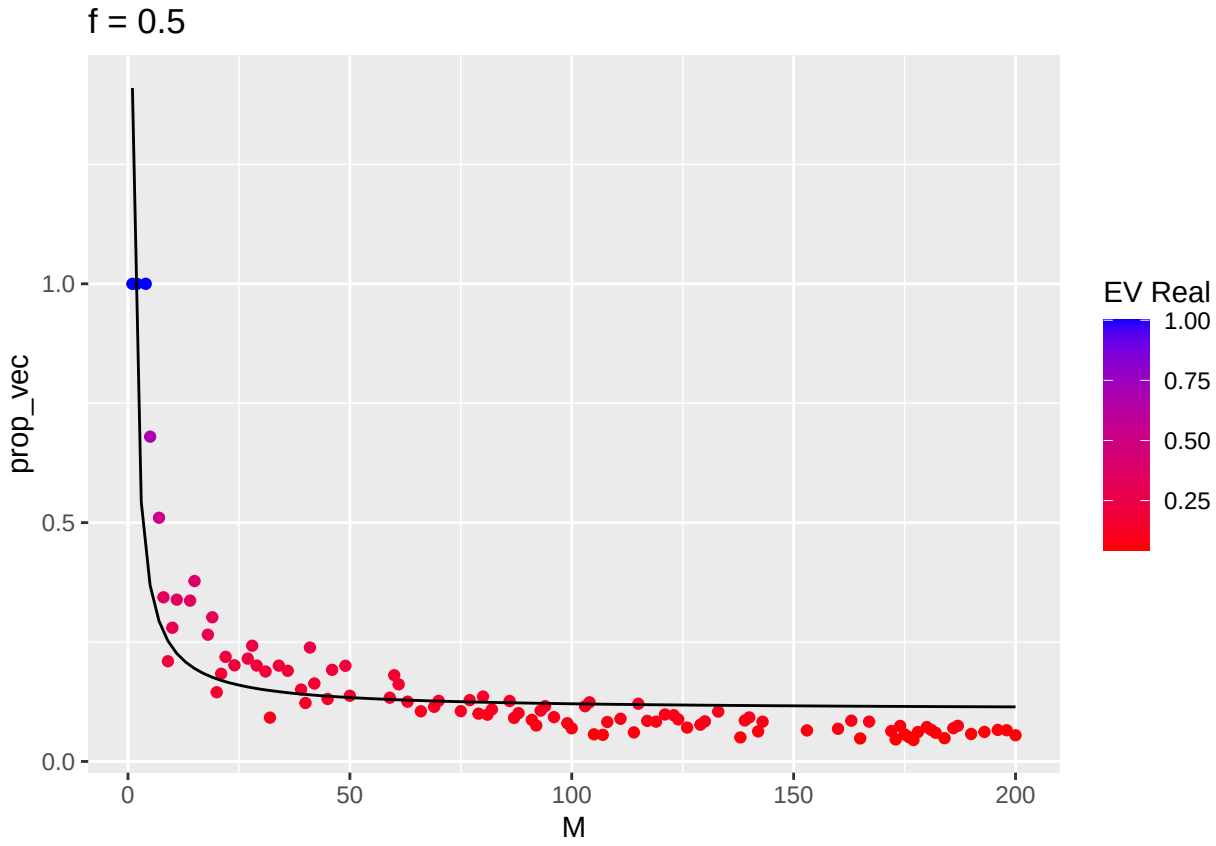
```
##     M   prop_vec
## 1   2 1.00000000
## 2 114 0.06092544
## 3 175 0.05683771
## 4 190 0.05760842
## 5  11 0.33880909
## 6 196 0.06603980
```

```
model <- glm(formula = prop_vec ~ I(1/M), data = table)
summary(model)
```

```
##
## Call:
## glm(formula = prop_vec ~ I(1/M), data = table)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.40991  -0.04726  -0.02449   0.01656   0.56668
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10779    0.01042   10.34   <2e-16 ***
## I(1/M)       1.30212    0.08625   15.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.009697154)
##
##     Null deviance: 3.16053  on 99  degrees of freedom
## Residual deviance: 0.95032  on 98  degrees of freedom
## AIC: -175.82
##
## Number of Fisher Scoring iterations: 2
```

```
pred <- function(M){
  b0 <- model$coefficients[1]
  b1 <- model$coefficients[2]
  b0 + b1/M
}

ggplot() +
  geom_point(data = table, aes(x=M, y=prop_vec, color = prop_vec)) +
  labs(color = "EV Real", title = paste("f = ",f,sep="")) +
  scale_color_gradient(high="blue", low="red") +
  stat_function(mapping = aes(x = M), fun = pred)
```


f = 0.5
prop_vec
1.0
0.5
0.0
0
50
100
150
200
M
EV Real
1.00
0.75
0.50
0.25