Sparsity Analysis

Ali Taqi

Generating Random Matrices

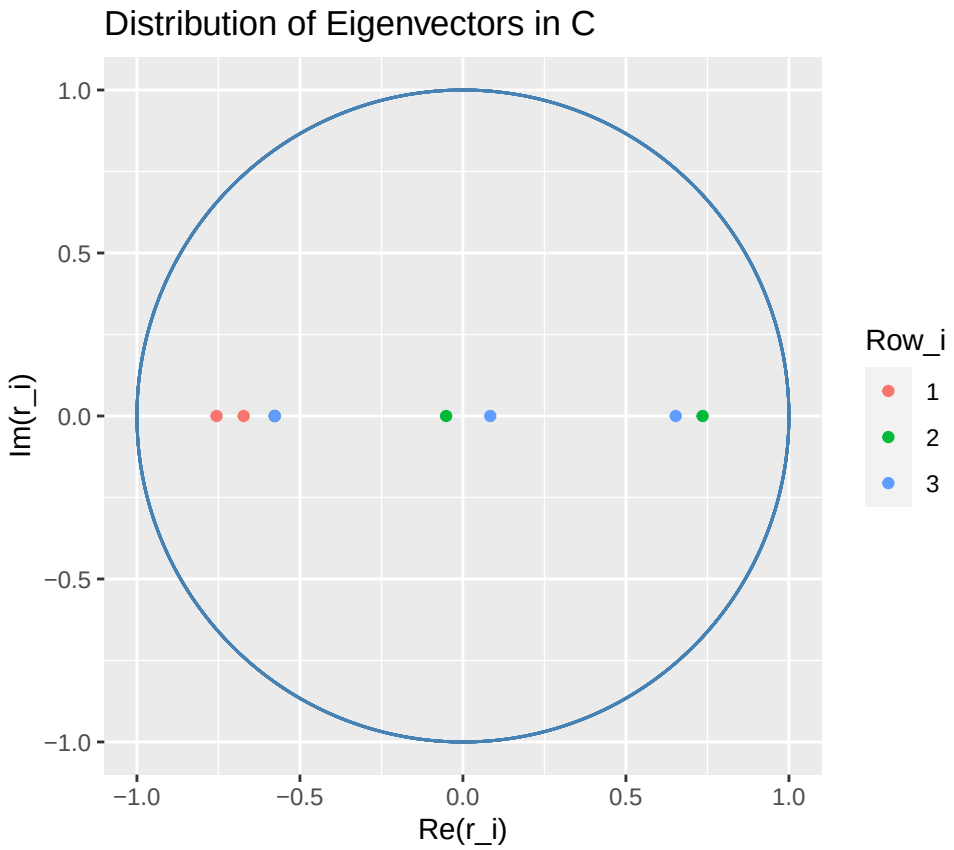
```
# generates rows of size P which are valid probability distributions
r_sparse <- function(M,p){
  prob <- runif(M,0,1)
  num_zeros <- rbinom(1,M,p)
  choices <- sample(1:M, num_zeros)
  prob[choices] <- 0
  prob/sum(prob) # return normalized random row vector
}
```

```
# initialize random P
rand_M <- function(M,p,row_fxn){
  P <- matrix(rep(NA, M * M), ncol = M)  # create transition matrix
  for(i in 1:M){P[i,] = row_fxn(M,p)}
  #print(P)
  P
}
```

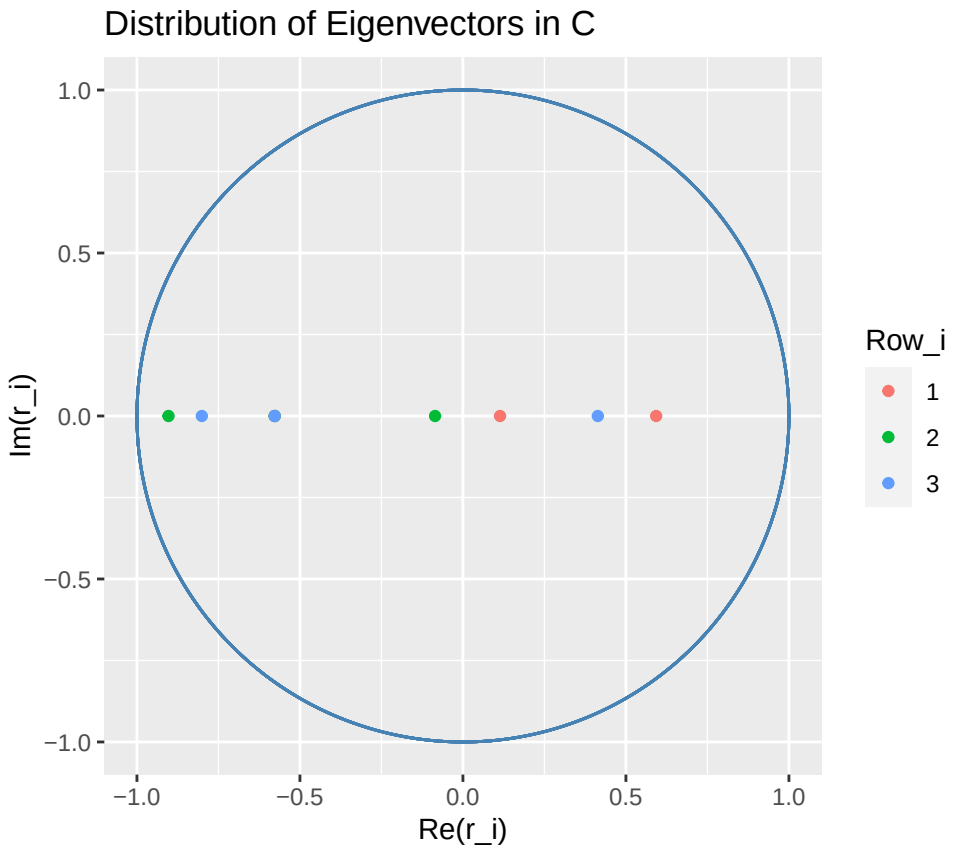
Eigenvectors

```
eigen_frame <- function(P){
  #print(P)
  M <- length(P[1,])
  eigenvectors <- data.frame(eigen(P)[2])
  complex <- matrix(rep(NA,3*M*M), ncol = 3) # set 3 to hold (re,im) pair and whose row it belongs to
  colnames(complex) <- c("Re","Im","row_i")
  for(i in 1:M){
    for(j in 1:M){
      curr <- eigenvectors[i,j]
      complex[ M*(i-1) + j, ] <- c(round(Re(curr),5),round(Im(curr),5),i)
    }
  }
  data.frame(complex)
}
```

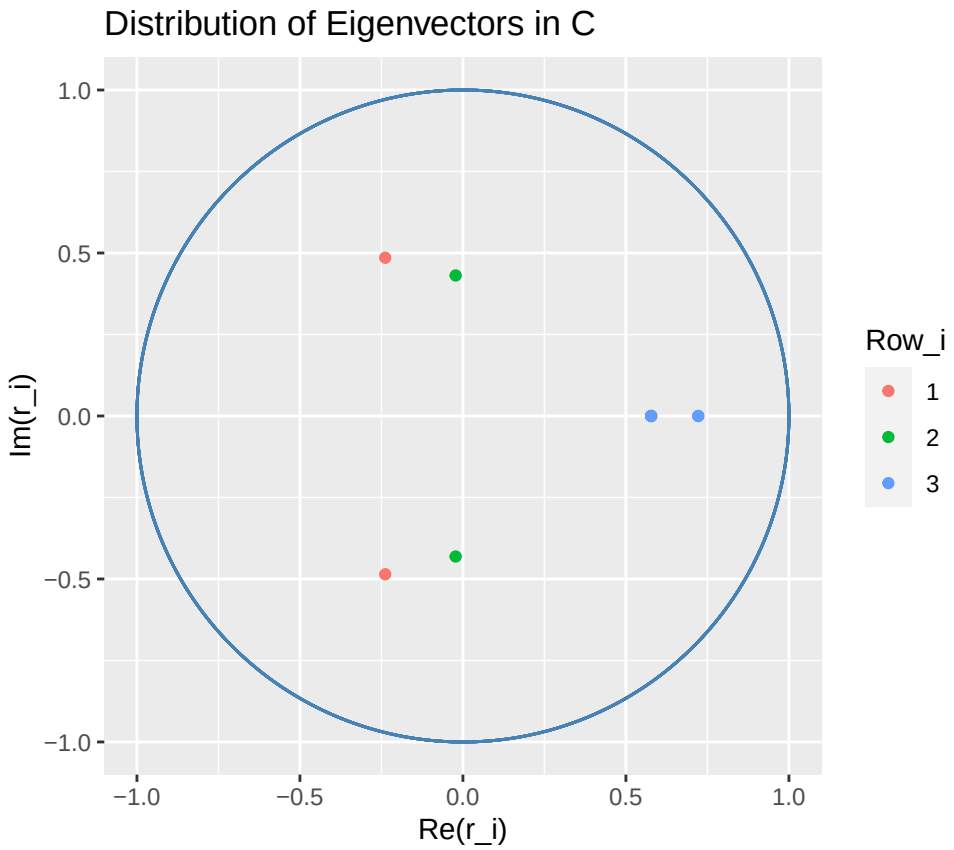
```
M_vec <- c(3,5,10)
p_vec <- c(0.1,0.5,0.6)
c(M1,M2,M3) %<-% M_vec
c(p1,p2,p3) %<-% p_vec
P_vec1 <- matrix(c(rand_M(M1,p1,r_sparse),
                   rand_M(M1,p1,r_sparse),
                   rand_M(M1,p1,r_sparse)),
                 nrow = M_vec[1])
P_vec2 <- matrix(c(rand_M(M2,p2,r_sparse),
                   rand_M(M2,p2,r_sparse),
                   rand_M(M2,p2,r_sparse)),
                 nrow = M_vec[2])
P_vec3 <- matrix(c(rand_M(M3,p3,r_sparse),
                   rand_M(M3,p3,r_sparse),
                   rand_M(M3,p3,r_sparse)),
                 nrow = M_vec[3])

eigen_plot(P_vec1[,1:M1 + 0*M1])
```



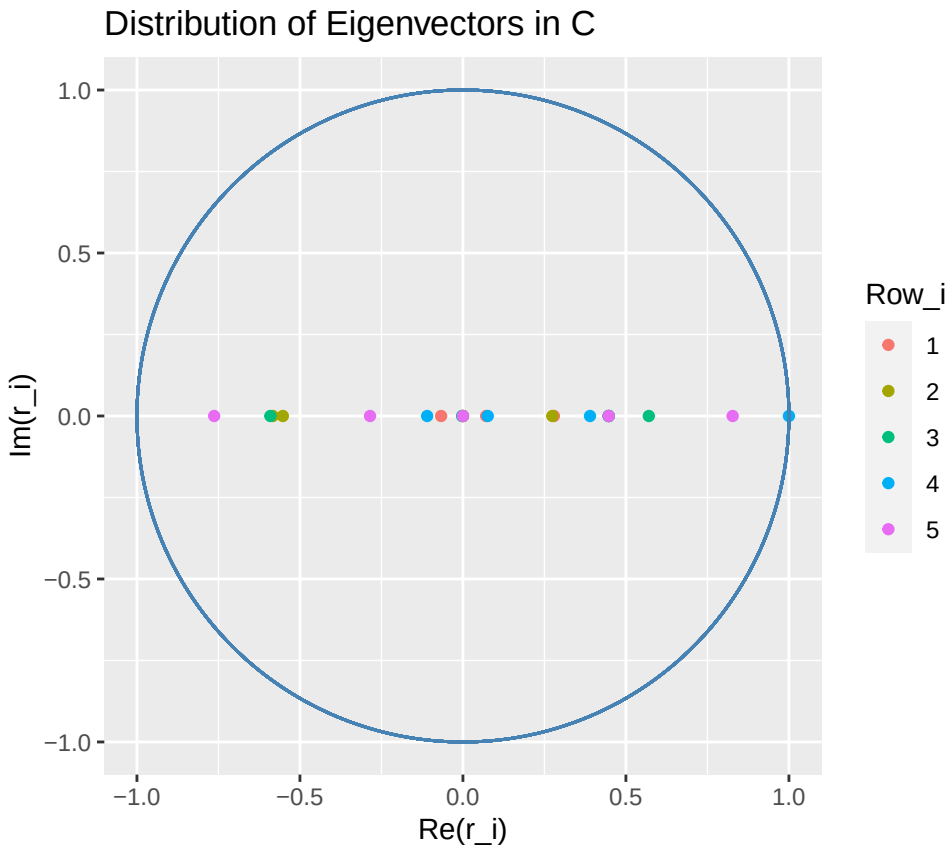

```
eigen_plot(P_vec1[,1:M1 + 1*M1])
```



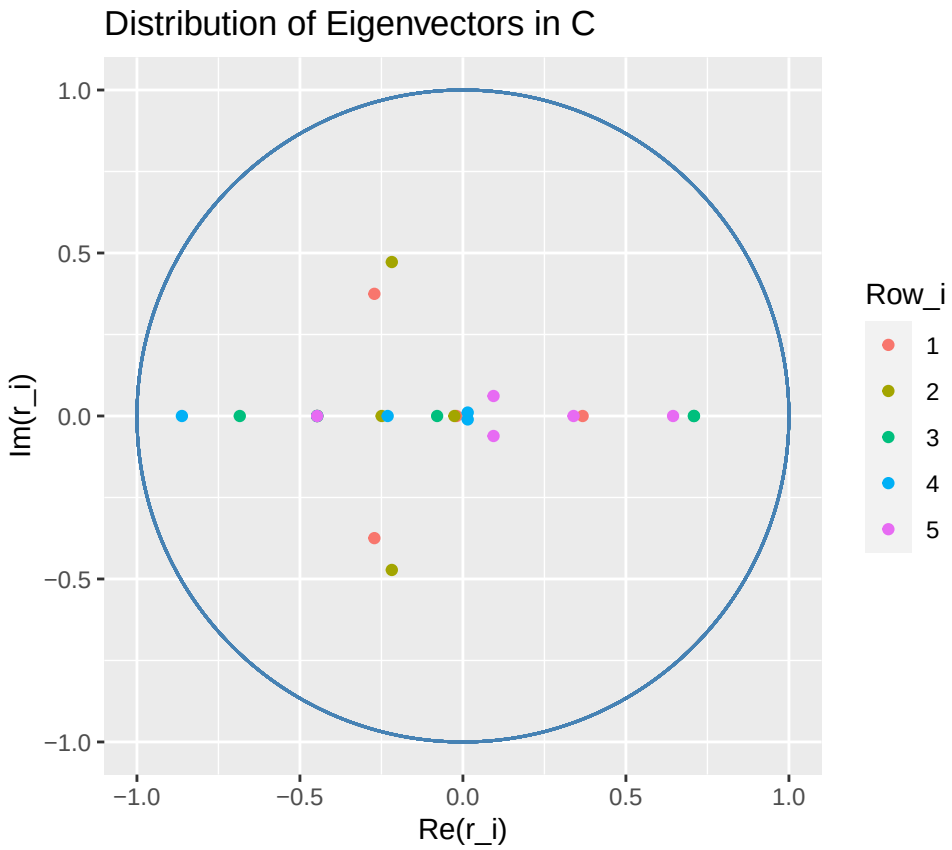

```
eigen_plot(P_vec1[,1:M1 + 2*M1])
```



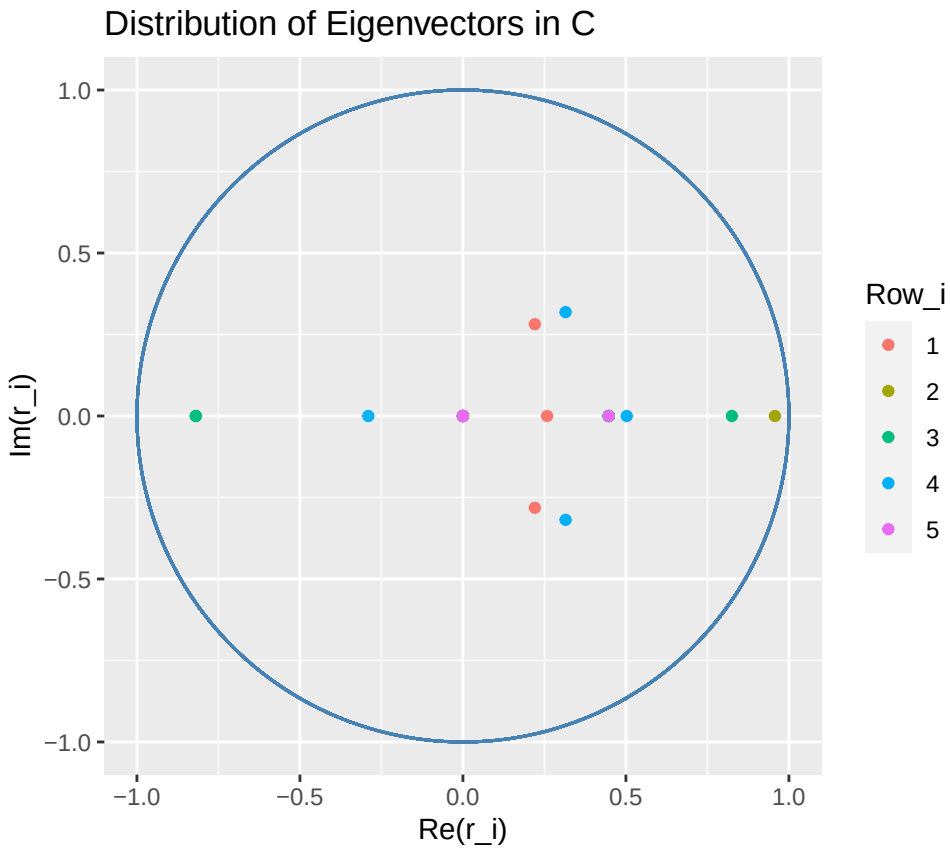

```
eigen_plot(P_vec2[,1:M2 + 0*M2])
```



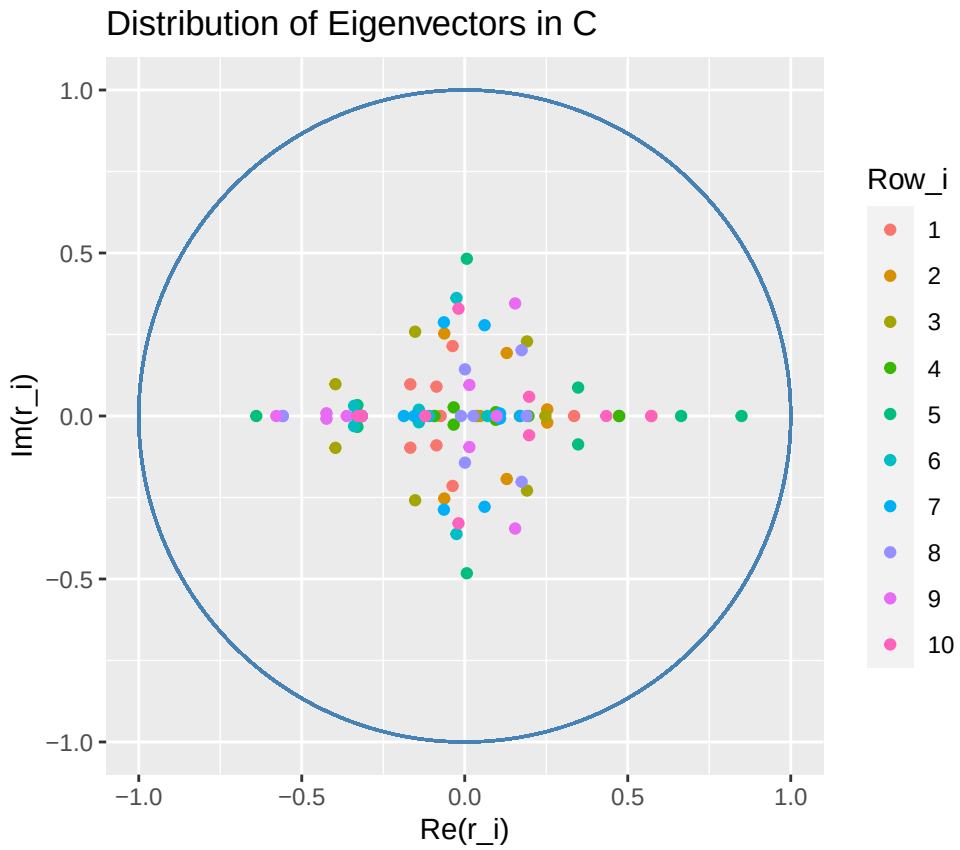

```
eigen_plot(P_vec2[,1:M2 + 1*M2])
```



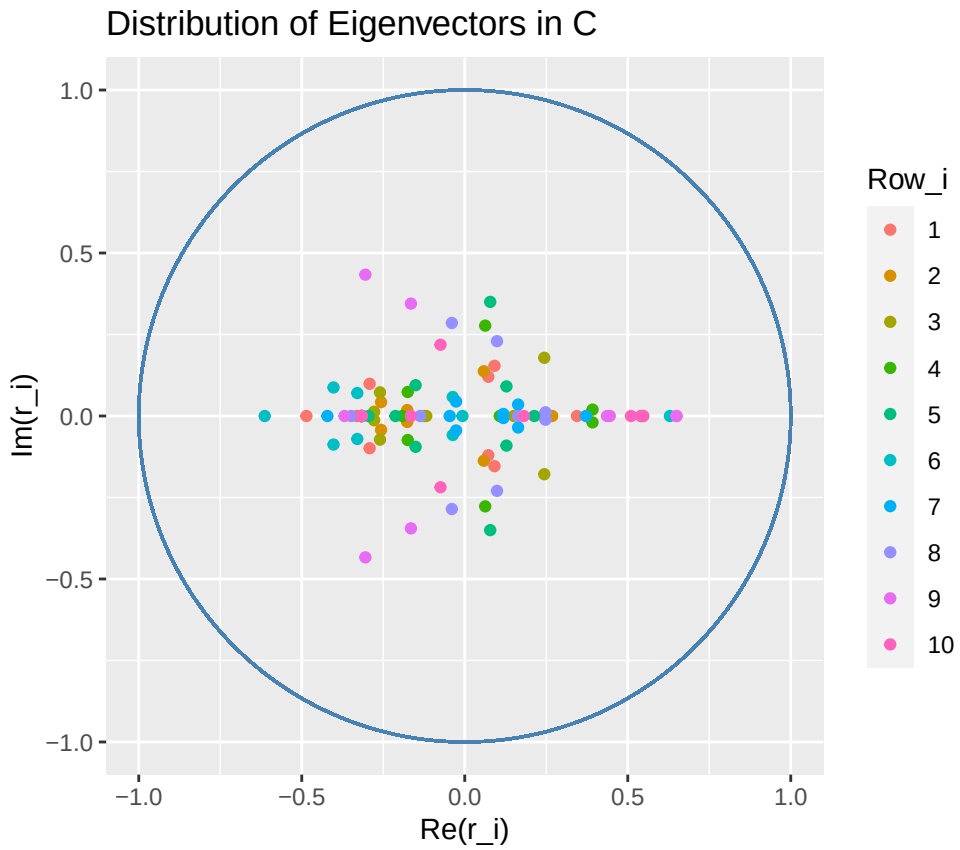

```
eigen_plot(P_vec2[,1:M2 + 2*M2])
```



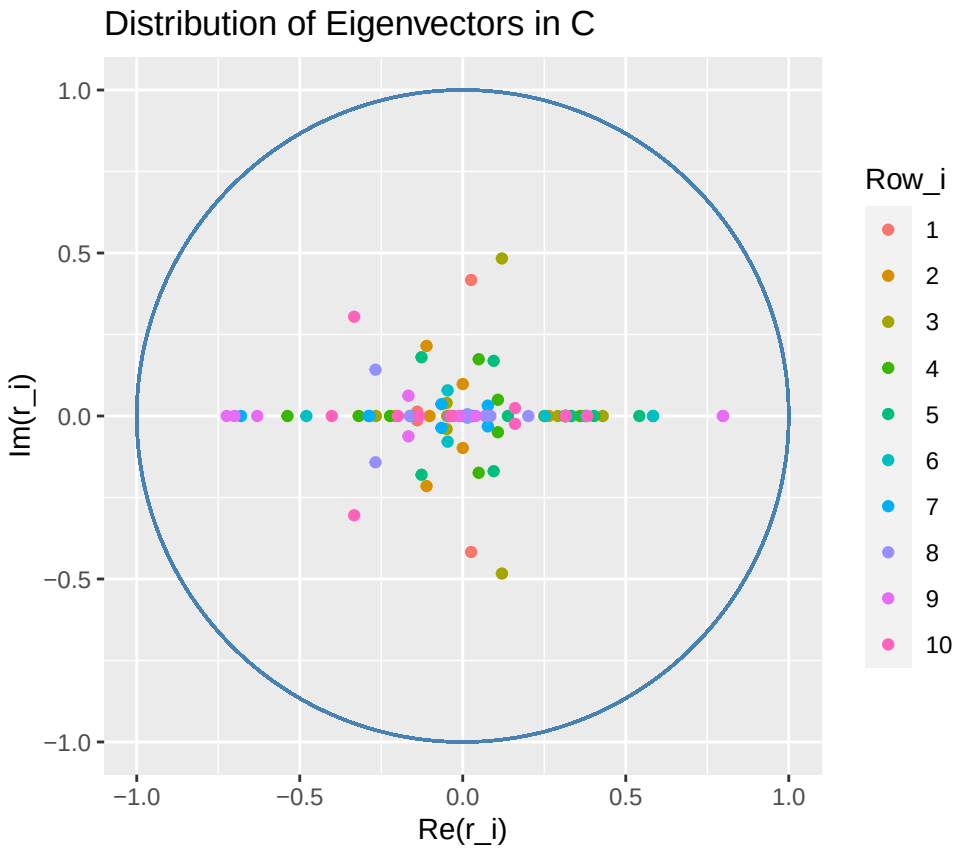

```
eigen_plot(P_vec3[,1:M3 + 0*M3])
```




```
eigen_plot(P_vec3[,1:M3 + 1*M3])
```




```
eigen_plot(P_vec3[,1:M3 + 2*M3])
```


Distribution of Eigenvectors in C
1.0
0.5
0.0
−0.5
−1.0
Im(r_i)
−1.0
−0.5
0.0
0.5
1.0
Re(r_i)
Row_i
1
2
3
4
5
6
7
8
9
10