202C - HW1

Jiayu Wu

2018/4/29

Problem 1

1

K_0 is irreducible with paths connecting each two nodes, and it is aperiodic with period d=1.

K_1 is not irreducible since node 5 and 1 are not connected, and aperiodic because every state is aperiodic.

K_2 is irreducible, and not aperiodic with period d=2.

2

```
K0 = matrix(c(.2, .8, .0, .0, .0,
              .3, .0, .4, .0, .3,
              .1, .4, .0, .5, .0,
              .0, .0, .4, .2, .4,
              .0, .0, .3, .6, .1),byrow = T, nrow = 5)
K1 = matrix(c(.3, .1, .3, .0, .3,
              .2, .4, .4, .0, .0,
              .0, .6, .3, .1, .0,
              .0, .0, .0, .5, .5,
              .0, .0, .0, .4, .6),byrow = T, nrow = 5)
K2 = matrix(c(.0, .0, .0, .3, .7,
              .0, .0, .0, .5, .5,
              .0, .0, .0, .8, .2,
              .0, .4, .6, .0, .0,
              .5, .0, .5, .0, .0),byrow = T, nrow = 5)
save(K0, file = "kernels.RData")
vals0 = eigen(K0)$values
vals1 = eigen(K1)$values
vals2 = eigen(K2)$values
vecs0 = eigen(t(K0))$vectors
vecs1 = eigen(t(K1))$vectors
vecs2 = eigen(t(K2))$vectors
d = cbind(cbind(rbind(vals0, vecs0), rbind(vals1, vecs1)), rbind(vals2, vecs2))
colnames(d) = c('K0_1','K0_2','K0_3','K0_4','K0_5','K1_1','K1_2','K1_3','K1_4','K1_5','K2_1','K2_2','K2_3','K2_4'
,'K2_5')
rownames(d) = c('values','vectors','','','','')
knitr::kable(d, digits = 3, main="Eigen values and vectors")
```

	K0_1	K0_2	K0_3	K0_4	K0_5	K1_1	K1_2	K1_3	K1_4	K1_5	K2_1	K2_2	K2_3	K2_4	K2_5
values	1.000	-0.615	0.529	-0.322	-0.092	1.000	0.916	0.133	0.100	-0.049	1.000	-1.000	-0.361	0.361	0.000
vectors	-0.204	0.092	0.457	0.165	0.333	0.000	0.161	0.185	0.000	0.411	0.189	0.189	0.606	-0.606	0.487
	-0.371	-0.388	0.559	-0.242	-0.110	0.000	0.497	-0.155	0.000	-0.718	0.259	0.259	-0.485	0.485	-0.811
	-0.519	0.413	-0.175	-0.135	-0.641	0.000	0.401	0.038	0.000	0.469	0.578	0.578	-0.121	0.121	0.324
	-0.625	-0.634	-0.638	0.767	-0.226	-0.625	-0.469	-0.719	-0.707	0.121	0.649	-0.649	0.437	0.437	0.000
	-0.401	0.518	-0.204	-0.555	0.644	-0.781	-0.589	0.651	0.707	-0.283	0.377	-0.377	-0.437	-0.437	0.000

3.

For K_0, there is one: $(0.096, 0.175, 0.245, 0.295, 0.189)$

For K_1, there is one: $(0, 0, 0, 0.444, 0.556)$

For K_2, there are two: $(0.092, 0.126, 0.282, 0.316, 0.184)$

4.

```r
load("kernels.RData")
power = diag(rep(1,5))
for (i in 1:200) {
      power = power %*% K0
}
power
```

```
##            [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.09613984 0.1747997 0.2447196 0.2949745 0.1893664
## [2,] 0.09613984 0.1747997 0.2447196 0.2949745 0.1893664
## [3,] 0.09613984 0.1747997 0.2447196 0.2949745 0.1893664
## [4,] 0.09613984 0.1747997 0.2447196 0.2949745 0.1893664
## [5,] 0.09613984 0.1747997 0.2447196 0.2949745 0.1893664
```

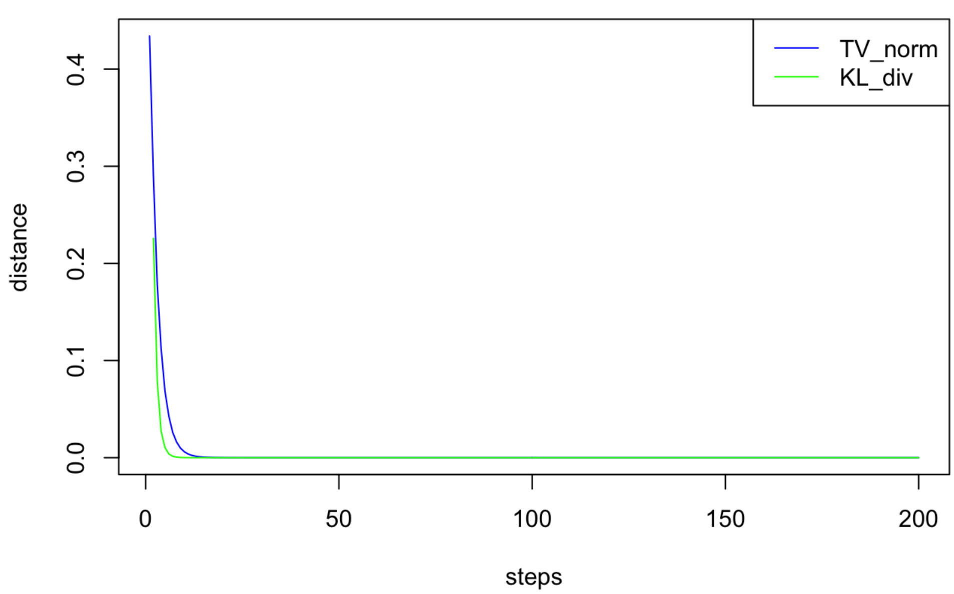
```r
pi = power[1,]
save(K0, pi, file = "kernels.RData")
```

Yes, it does.

Problem 2

1.

```r
load("kernels.RData")
v = c(0, 0, 1, 0, 0)
tv = c()
kl = c()
for (i in 1:200){
       v = v %*% K0
       tv[i] = 0.5*sum(abs(pi-v))
       kl[i] = sum(pi*log(pi/v))
}
plot(1:200, tv, type = "l", col="blue", xlab = "steps", ylab = "distance")
lines(1:200, kl, col = "green")
legend("topright", legend=c("TV_norm", "KL_div"), col = c("blue", "green"), lty = 1)
```



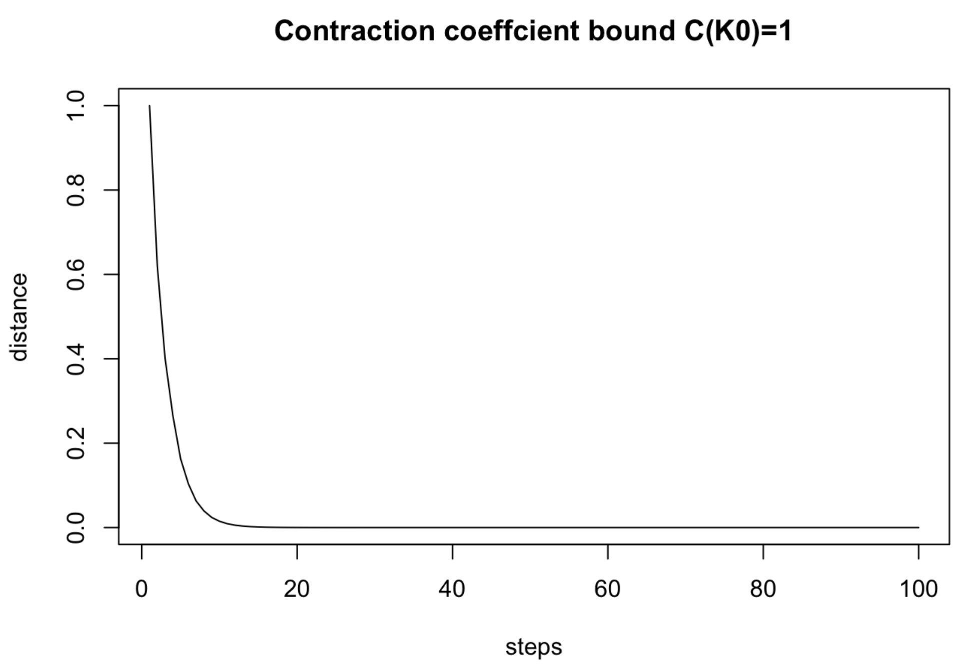

```r
save(K0, pi, tv, file = "kernels.RData")
```

2.

```r
load("kernels.RData")
tvs = matrix(rep(0,25), nrow = 5)
for (i in 1:5) {
	for (j in i:5) {
		tvs[i,j] = 0.5*sum(abs(K0[i,]-K0[j,]))
	}
}
max(tvs)
```

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

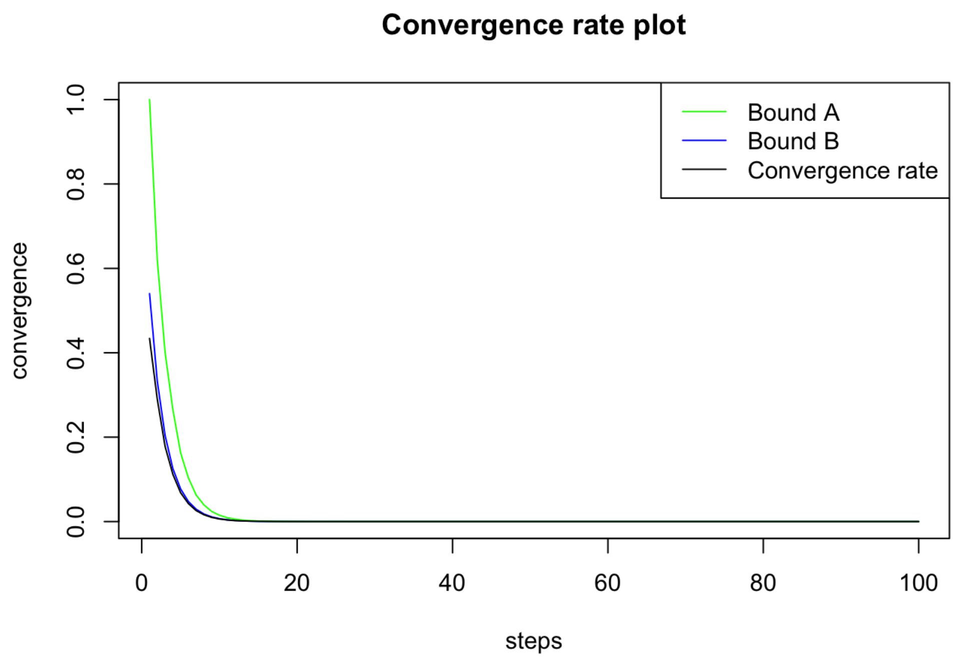
```r
# bound
boundsa = c()
power = diag(rep(1,5))
for (k in 1:100) {
	power = power %*% K0
	tvs = matrix(rep(0,25), nrow = 5)
	for (i in 1:5) {
		for (j in i:5) {
			tvs[i,j] = 0.5*sum(abs(power[i,]-power[j,]))
		}
	}
	boundsa[k] = max(tvs)
}
plot(1:100, boundsa, type = "l", xlab = "steps", ylab = "distance",
     main = "Contraction coeffcient bound C(K0)=1")
```




```r
save(K0, pi, tv, boundsa,  file = "kernels.RData")
```

3.

```r
load("kernels.RData")
sq = sqrt((1-pi[3])/pi[3]/4)
boundsb = apply(t(1:100),2,function(x){sq*(0.61504869^x)})
plot(1:100, boundsb, type = "l", col = "blue", xlab = "steps",
     ylab = "convergence", ylim=c(0,1), main = "Convergence rate plot")
lines(1:100, boundsa, col="green")
lines(1:100, tv[1:100], col="black")
legend("topright", legend=c("Bound A", "Bound B", "Convergence rate"), col = c("green", "blue", "black"), lty = 1
)
```




```
# log-plot
plot(1:20, log(boundsb[1:20]), type = "l", col = "blue", xlab = "steps",
     ylab = "convergence", main = "Convergence rate zoom-in log-plot")
lines(1:20, log(boundsa[1:20]), col="green")
lines(1:20, log(tv[1:20]), col="black")
legend("topright", legend=c("Bound A", "Bound B", "Convergence rate"), col = c("green", "blue", "black"), lty = 1
)
```

Problem 3

$$
\begin{aligned}
KL(\pi||v) - KL(\pi||\mu) &= KL(\pi(x)\mu(y)||\pi(y)\mu(x)) = \sum_x \sum_y P(y,x) log \frac{\pi(x)\mu(y)}{\pi(y)v(x)} \\
&= \sum_x \sum_y P(y,x) log \frac{\mu(y)P(y,x)}{v(x)P(x,y)} \\
&= \sum_x \sum_y P(y,x) log \frac{P(y,x)}{Q(y,x)} \\
&= E[KL(P(y,x)||Q(y,x))]
\end{aligned}
$$

$$
\begin{aligned}
\because P(y,x) &= \frac{\pi(x)P(x,y)}{\pi(y)} = \frac{\mu(y)\pi(x)}{v(x)\pi(y)} \\
KL(P(y,x)||P(y,x)) &= \sum_x \sum_y \frac{\pi(x)P(x,y)}{\pi(y)} log \frac{\pi(x)P(x,y)v(x)\pi(y)}{\pi(y)\mu(y)\pi(x)} \\
&= \sum_x \sum_y P(y,x) log \frac{P(x,y)v(x)}{\mu(y)} \\
&= \sum_x \sum_y P(y,x) log Q(y,x) = 0 \\
E[KL(P(y,x)||Q(y,x))] &= \sum_x \sum_y P(y,x) log \frac{P(y,x)}{Q(y,x)} \\
&= \sum_x \sum_y P(y,x) log P(y,x) - \sum_x \sum_y P(y,x) log Q(y,x) \\
&= E_{P(y,x)}[log P(y,x)] \geq 0
\end{aligned}
$$

Problem 4

$$
P(\tau(0) < \infty) = \lim_{n\to\infty} \sum_{\tau(0)=1}^{n} P(\tau(0)) = \lim_{n\to\infty} \sum_{\tau(0)=1}^{n} \alpha^{\tau(0)-1}(1-\alpha) = \lim_{n\to\infty}(1-\alpha)\frac{1-\alpha^{n-1}}{1-\alpha} = 1
$$

$$
E[\tau_{ret}(0)] = \lim_{n\to\infty} \sum_{\tau(0)=1}^{n} \tau(0)P(\tau(0)) = \lim_{n\to\infty} \sum_{\tau(0)=1}^{n} \tau(0)\alpha^{\tau(0)-1}(1-\alpha) = (1-\alpha) \lim_{n\to\infty}\left(\frac{1-\alpha^n}{(1-\alpha)^2} + \frac{n\alpha^n}{1-\alpha}\right) = (1-\alpha)\frac{1}{(1-\alpha)^2} = \frac{1}{1-\alpha}
$$