Colas.pdf

CintiaOjeda

Ampliación de Investigación Operativa

3º Grado en Estadística

Facultad de Matemáticas
Universidad de Sevilla

Teoría de colas

Cintia Mª Ojeda Silva

2022-05-27

INTRODUCCIÓN

```
library(queueing)
```

Le damos los parámetros

```
i_mm1<-NewInput.MM1(lambda=8,mu=12)
i_mm1
```

```
## $lambda
## [1] 8
##
## $mu
## [1] 12
##
## $n
## [1] 0
##
## attr(,"class")
## [1] "i_MM1"
```

Comprobamos que está bien (si no dice nada es que no hay error)

```
CheckInput(i_mm1)
```

Creamos la cola

```
q_mm1<-QueueingModel(i_mm1)

summary(q_mm1)
```

```
##   lambda mu c  k  m        RO        P0       Lq        Wq X L    W  Wqq Lqq
## 1      8 12 1 NA NA 0.6666667 0.3333333 1.333333 0.1666667 8 2 0.25 0.25   3
```

MODELO DE COLAS SIMPLES.

Es habitual admitir que los clientes llegan/son atendidos con una distribución Poisson (lambda) y la duración de la atención sigue una distribución exponencial (mu).

M/M/1

6.1 Encargado de Bibliotecas.

Un estudiante trabaja como encargado de una biblioteca por las noches y es el único en el mostrador durante todo su turno de trabajo. Las llegadas al mostrador siguen una distribución de Poisson con una media de 8 por hora. Cada usuario de la biblioteca es atendido de uno en uno, y el tiempo de servicio sigue una distribución exponencial con una media de 5 minutos.

lambda=8 clientes/h mu=12 clientes atendidos/h

```
i_mm6.1=NewInput.MM1(lambda=8,mu=12)
q_mm6.1=QueueingModel(i_mm6.1)
summary(q_mm6.1)
```

```
##   lambda mu c  k  m        RO        P0       Lq        Wq X L    W  Wqq Lqq
## 1      8 12 1 NA NA 0.6666667 0.3333333 1.333333 0.1666667 8 2 0.25 0.25   3
```

a)Probabilidad de que se forme cola (o lo que es lo mismo, 1- Pr(no se forme cola)=1-P0).

```
1-q_mm6.1$Pn
```

```
## [1] 0.6666667
```

b)Longitud media de la cola.

```
q_mm6.1$Lq
```

```
## [1] 1.333333
```

c)Tiempo medio hasta ser atendido

```
q_mm6.1$W #el resultado está en horas
```

```
## [1] 0.25
```

d)Tiempo medio en la cola

```
q_mm6.1$Wq
```

```
## [1] 0.1666667
```

e)Cantidad de fichas que puede ordenar. Suponemos que tine una jornada de 8 horas. Podrá ordenar siempre que no tenga clientes, luego:

FIEP

FERIA INTERNACIONAL DE ESTUDIOS DE POSTGRADO

2023

FIEP - Feria de Postgrado

Sevilla

13 MAR. | Hotel NH Collection Sevilla

¡Inscríbete!
Te esperamos :)

- **Entrada Gratuita**
  para el acceso a FIEP
- **Sorteo de 3 Becas**
  de 10.000 € para tu postgrado
- **Más de 1.000.000€**
  para Becas y Ayudas
- **Salidas profesionales**
  y más información sobre tu máster
- **Presencial y Online**
  de Másters, Postgrados, Doctorados...

www.fiep.es

```r
q_mm6.1$Pn*8*22
```

```
## [1] 58.66667
```

6.2 Mantenimiento de Coches.

Una compañía de alquiler de coches tiene un servicio de mantenimiento de coches (revisión del aceite, frenos, lavado…) que sólo es capaz de atender los coches de uno en uno y que trabaja 24 horas al día. Los coches llegan al taller con una media de 3 coches por día. El tiempo que dura el servicio de mantenimiento de un coche sigue una distribución exponencial de media 7 horas. El servicio de mantenimiento cuesta a la compañía 375 euros por día. La compañía estima en 25 euros/día el coste de tener el coche parado sin poderse alquilar. La compañía se plantea la posibilidad de cambiar el servicio de mantenimiento por uno más rápido que puede bajar el tiempo de mantenimiento a una media de 5 horas, pero esto también supone un incremento del coste. ¿Hasta que valor puede aumentar el coste para que la compañía contrate los nuevos servicios de mantenimiento?

mu=24/7=3.42857coches/día. lambda=3coches servidos/día.

```r
i_mm2<-NewInput.MM1(lambda=3,mu=24/7)
q_mm2<-QueueingModel(i_mm2)

summary(q_mm2)
```

```
##   lambda       mu c  k  m    RO    P0    Lq       Wq X L        W      Wqq Lqq
## 1      3 3.428571 1 NA NA 0.875 0.125 6.125 2.041667 3 7 2.333333 2.333333   8
```

No perdemos nada mientras están en el sistema (están siendo arreglados y no se pueden alquilar)

```r
coste_m2<-25*q_mm2$L
cat("El coste es = ", coste_m2,"\n")
```

```
## El coste es =  175
```

```r
coste_total_m2<-375+coste_m2
cat("El coste total (con el mantenimiento) es = ",coste_total_m2,"\n")
```

```
## El coste total (con el mantenimiento) es =  550
```

Otro modelo de colas (si tardan 5 horas)

```r
i_mm3<-NewInput.MM1(lambda = 3,mu=24/5)
q_mm3<-QueueingModel(i_mm3)
summary(q_mm3)
```

```
##   lambda  mu c  k  m    RO    P0       Lq        Wq X        L         W
## 1      3 4.8 1 NA NA 0.625 0.375 1.041667 0.3472222 3 1.666667 0.5555556
##         Wqq      Lqq
## 1 0.5555556 2.666667
```

```
coste_m3<-25*q_mm3$L
maximo_coste_m3<-coste_total_m2-coste_m3
cat("El coste máximo es = ",maximo_coste_m3)
```

```
## El coste máximo es =  508.3333
```

M/M/C

6.3 Comidas Rápidas.

Nuestro local de comida rápida, "Panis", tiene mucho que aprender sobre teoría de colas. Insta a los clientes a que formen 3 colas en las que se distribuyen de forma aleatoria delante de los empleados durante el periodo de comidas diario. Además han instalado entre las tres colas barreras para que los clientes no se pasen a otras colas para prevenir que la gente se "cambie de cola". Llegan los clientes según una distribución de Poisson con una media de 60 por hora y el tiempo en que un cliente es servido varía según una distribución exponencial de media 150 segundos. Asumiendo el estado permanente del sistema, ¿cuál es el tiempo medio de estancia del cliente hasta que ha sido atendido? El gerente de "Panis" ha creído ahora que es preferible una única cola para distribuir finalmente a los tres servidores y por tanto las barreras son eliminadas. ¿cuál es el tiempo de espera de este modo?

Datos: c=3 (hay 3 colas). mu=24 (3600/150) clientes/hora. lambda=60 clientes/hora.

```
i_mmc<-NewInput.MMC(lambda=60,mu=24, c=3,n=10) #n arbitrario
q_mmc<-QueueingModel(i_mmc)                    #prob(0 a 10)
summary(q_mmc)
```

```
##   lambda mu c  k  m        RO         P0       Lq        Wq  X        L
## 1     60 24 3 NA NA 0.8333333 0.04494382 3.511236 0.0585206 60 6.011236
##          W        Wqq Lqq
## 1 0.1001873 0.08333333   6
```

Probabilidad de que haya 0,1,2…10 personas.

```
q_mmc$Pn
```

```
##  [1] 0.04494382 0.11235955 0.14044944 0.11704120 0.09753433 0.08127861
##  [7] 0.06773218 0.05644348 0.04703623 0.03919686 0.03266405
```

```
cat("La probabilidad de que haya 10 personas es = ", q_mmc$Pn[11],"\n","Cada uno puede atender a 24 cli
```

```
## La probabilidad de que haya 10 personas es =  0.03266405
##  Cada uno puede atender a 24 clientes
```

Con el triple de capacidad de servicio,1 cola 3 veces más rápida.

```
i_mmc2<-NewInput.MM1(lambda=60,mu=3*24,n=10)
q_mmc2<-QueueingModel(i_mmc2)
summary(q_mmc2)
```

```
##   lambda mu c  k  m        RO        P0       Lq         Wq  X L          W
## 1     60 72 1 NA NA 0.8333333 0.1666667 4.166667 0.06944444 60 5 0.08333333
##          Wqq Lqq
## 1 0.08333333   6
```

```
q_mmc2$Pn
```

```
##  [1] 0.16666667 0.13888889 0.11574074 0.09645062 0.08037551 0.06697960
##  [7] 0.05581633 0.04651361 0.03876134 0.03230112 0.02691760
```

```
cat("La probabilidad de que haya 10 personas es = ", q_mmc2$Pn[11])
```

```
## La probabilidad de que haya 10 personas es =  0.0269176
```

En el segundo tardamos 6 min de media en la cola, y en el primero 5 min (Wq). La probabilidad de que no haya ningún cliente es mayor en el 2º. En cambio, en el 2º estamos menos tiempo en el servicio (W).

```
(1-q_mmc2$W/q_mmc$W)*100 #% de ahorro de tiempo del cliente
```

```
## [1] 16.82243
```

6.4 Coordinación de transmisiones.

Una organización está actualmente envuelta en el establecimiento de un centro de telecomunicaciones para tener una mejor capacidad de las mismas. El centro deberá ser el responsable de la salida de los mensajes así como de la entrada y distribución dentro de la organización. El encargado del centro es el responsable de determinar los operadores que deben trabajar en él. Los operarios encargados de la salida de mensajes son responsables de hacer pequeñas correcciones a los mensajes, mantener un índice de códigos y un fichero con los mensajes salientes en los últimos 30 días, y por supuesto, transmitir el mensaje. Se ha establecido que este proceso es exponencial y requiere una media de 28 min/mensaje. Los operarios de transmisión trabajarán en el centro 7 horas al día y cinco días a la semana. Todos los mensajes salientes serán procesados según el orden en que se vayan recibiendo y siguen una distribución de Poisson con una media de 21 por cada 7 horas diarias. Los mensajes deben ser atendidos en 2 horas como máximo. Determine el número mínimo de personal que se necesita para cumplir este criterio de servicio.

Datos: lambda=21/7=3 mensajes/hora. mu=60/28 mensajes producidos/hora (algo más de 2).

```
i_mm5<-NewInput.MM1(lambda=3,mu=60/28)
i_mm5
```

```
## $lambda
## [1] 3
##
## $mu
## [1] 2.142857
##
## $n
## [1] 0
##
## attr(,"class")
## [1] "i_MM1"
```

Comprobamos que está bien: Ejecutamos CheckInput(i_mm5) y obtenemos que rho es mayor que 1.

Con 2 colas:

```
i_mmc5<-NewInput.MMC(lambda=3,mu=60/28,c=2,n=0)
q_mmc5<-QueueingModel(i_mmc5)
summary(q_mmc5)
```

```
##   lambda       mu c  k  m  RO        P0       Lq       Wq X        L         W
## 1      3 2.142857 2 NA NA 0.7 0.1764706 1.345098 0.448366 3 2.745098 0.9150327
##         Wqq      Lqq
## 1 0.7777778 3.333333
```

```
Report(q_mmc5)
```

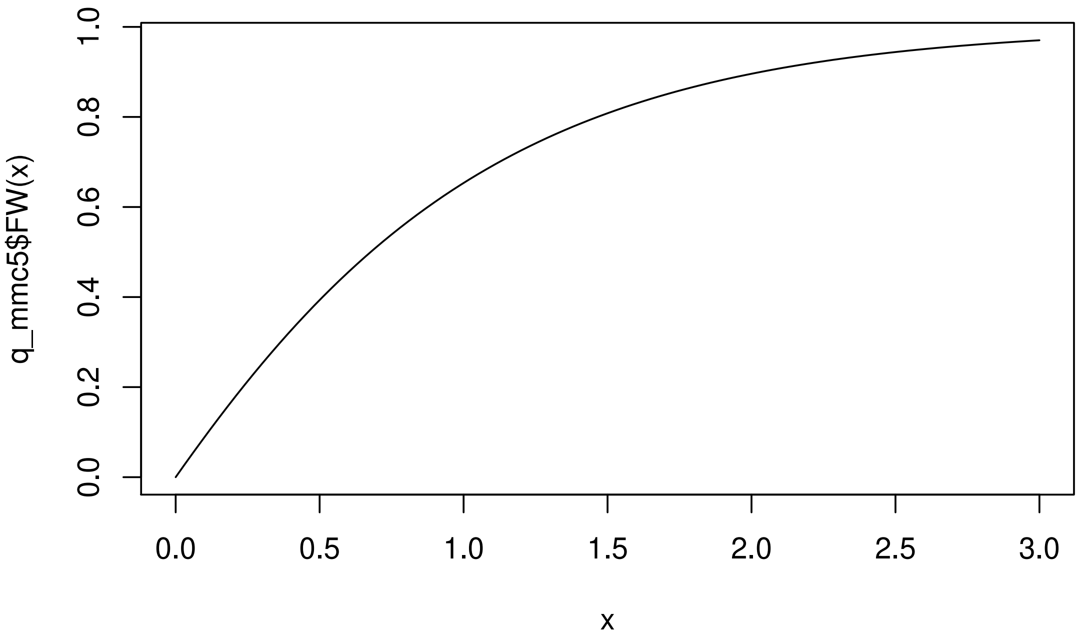
```
## The inputs of the model M/M/c are:
## lambda: 3, mu: 2.14285714285714, c: 2, n: 0, method: Exact
##
## The outputs of the model M/M/c are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.1764706
## The traffic intensity is: 1.4
## The server use is: 0.7
## The mean number of clients in the system is: 2.74509803921569
## The mean number of clients in the queue is: 1.34509803921569
## The mean number of clients in the server is: 1.4
## The mean time spend in the system is: 0.915032679738562
## The mean time spend in the queue is: 0.448366013071896
## The mean time spend in the server is: 0.466666666666667
## The mean time spend in the queue when there is queue is: 0.777777777777778
## The throughput is: 3
```

Que los mensajes sean atendidos en menos de 2 horas:

```
cat("La probabilidad de que le mensaje sea atendido en menos de 2 horas es = ",q_mmc5$FW(2))
```

```
## La probabilidad de que le mensaje sea atendido en menos de 2 horas es =  0.8959285
```

```
curve(q_mmc5$FW(x),0,3)
```



Si probamos con c=3, la probabilidad de ser atendido en menos de dos horas asciende a 0,9819, y con c=4 esta probabilidad es de 0,9857.

6.5 Sucursal Bancaria.

Una pequeña sucursal de un banco tiene dos empleados, uno para los pagos y otro para los cobros. Los clientes llegan a cada caja siguiendo una distribución de Poisson con una media de 20/hora. (el total de llegada al banco es de 40/hora). El tiempo de servicio de cada empleado es una negativa exponencial de media 2 minutos. El encargado de la sección está pensando hacer un cambio en que los dos operarios puedan hacer tanto pagos como cobros para evitar situaciones en que una cola está llena y la otra parada. Sin embargo, se estima que cuando los empleados se encarguen de las dos cosas el tiempo de servicio aumentará a una media de 2,4 minutos. Compara el sistema que se emplea ahora con el propuesto, calculando el total de gente en el banco, el tiempo medio que pasaría un cliente en el banco hasta que es atendido, la probabilidad de que un cliente espere

Datos: lambda=40/hora. mu=30clientes/h (2 min por cliente).

```
i_mm6.5<-NewInput.MMC(lambda=40,mu=30,c=2)
```

Comprobamos que está bien

```
CheckInput(i_mm6.5)
```

Creamos la cola

```r
i_mmc6.5<-NewInput.MMC(lambda=40,mu=30,c=2)
q_mmc6.5<-QueueingModel(i_mmc6.5)
summary(q_mmc6.5)
```

```
##   lambda mu c  k  m        RO  PO       Lq         Wq  X   L    W  Wqq Lqq
## 1     40 30 2 NA NA 0.6666667 0.2 1.066667 0.02666667 40 2.4 0.06 0.05   3
```

```r
Report(q_mmc6.5)
```

```
## The inputs of the model M/M/c are:
## lambda: 40, mu: 30, c: 2, n: 0, method: Exact
##
## The outputs of the model M/M/c are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.2
## The traffic intensity is: 1.33333333333333
## The server use is: 0.666666666666667
## The mean number of clients in the system is: 2.4
## The mean number of clients in the queue is: 1.06666666666667
## The mean number of clients in the server is: 1.33333333333333
## The mean time spend in the system is: 0.06
## The mean time spend in the queue is: 0.0266666666666667
## The mean time spend in the server is: 0.0333333333333333
## The mean time spend in the queue when there is queue is: 0.05
## The throughput is: 40
```

```r
cat("El nº medio de clientes en cola es ",q_mmc6.5$L)
```

```
## El nº medio de clientes en cola es  2.4
```

```r
cat("\nEl tiempo medio hasta ser atendido es ",q_mmc6.5$W, " horas")
```

```
##
## El tiempo medio hasta ser atendido es  0.06  horas
```

```r
cat("\nLa probabilidad de que pase más de 5 min en cola es ", 1-q_mmc6.5$FW(5/60))
```

```
##
## La probabilidad de que pase más de 5 min en cola es  0.25295
```

Datos (por separado): lambda=20/hora. mu=60/2.4 clientes/h.

```r
i_mm6<-NewInput.MM1(lambda=20,mu=60/2.4)
CheckInput(i_mm6)
i_mmc6<-NewInput.MMC(lambda=20,mu=60/2.4,c=2)
q_mmc6<-QueueingModel(i_mmc6)
summary(q_mmc6)
```

```
##   lambda mu c  k  m  RO        P0       Lq          Wq  X        L          W
## 1     20 25 2 NA NA 0.4 0.4285714 0.152381 0.007619048 20 0.952381 0.04761905
##          Wqq      Lqq
## 1 0.03333333 1.666667
```

```
Report(q_mmc6)
```

```
## The inputs of the model M/M/c are:
## lambda: 20, mu: 25, c: 2, n: 0, method: Exact
##
## The outputs of the model M/M/c are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.4285714
## The traffic intensity is: 0.8
## The server use is: 0.4
## The mean number of clients in the system is: 0.952380952380952
## The mean number of clients in the queue is: 0.152380952380952
## The mean number of clients in the server is: 0.8
## The mean time spend in the system is: 0.0476190476190476
## The mean time spend in the queue is: 0.00761904761904762
## The mean time spend in the server is: 0.04
## The mean time spend in the queue when there is queue is: 0.0333333333333333
## The throughput is: 20
```

```
cat("El nº medio de clientes en cola es ",q_mmc6$L)
```

```
## El nº medio de clientes en cola es  0.952381
```

```
cat("\nEl tiempo medio hasta ser atendido es ",q_mmc6$W, " horas")
```

```
##
## El tiempo medio hasta ser atendido es  0.04761905  horas
```

```
cat("\nLa probabilidad de que pase más de 5 min en cola es ", 1-q_mmc6$FW(5/60))
```

```
##
## La probabilidad de que pase más de 5 min en cola es  0.1730053
```

En el 2º modelo que hemos construido estamos menos tiempo en cola.

EJEMPLO DE LA NOTA (sobre el ej 6.5).

```
i_mu8=NewInput.MM1(lambda = 2,mu=3)
CheckInput(i_mu8)
q_mu8=QueueingModel(i_mu8)
Report(q_mu8)
```

```
## The inputs of the M/M/1 model are:
## lambda: 2, mu: 3, n: 0
```

```
##
## The outputs of the M/M/1 model are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.3333333
## The traffic intensity is: 0.666666666666667
## The server use is: 0.666666666666667
## The mean number of clients in the system is: 2
## The mean number of clients in the queue is: 1.33333333333333
## The mean number of clients in the server is: 0.666666666666667
## The mean time spend in the system is: 1
## The mean time spend in the queue is: 0.666666666666667
## The mean time spend in the server is: 0.333333333333333
## The mean time spend in the queue when there is queue is: 1
## The throughput is: 2
```

```
i_cola1=NewInput.MM1(lambda = 2,mu=3) #MM1 son las dos idénticas
cola1=QueueingModel(i_cola1)
summary(cola1) #Idéntico para pagos y cobros
```

```
##   lambda mu c  k  m        RO        P0       Lq        Wq X L W Wqq Lqq
## 1      2  3 1 NA NA 0.6666667 0.3333333 1.333333 0.6666667 2 2 1   1   3
```

Veamos ahora el modelo simplificado.

```
i_cola2=NewInput.MM1(lambda = 4,mu=6)
cola2=QueueingModel(i_cola2)
summary(cola2)
```

```
##   lambda mu c  k  m        RO        P0       Lq        Wq X L   W Wqq Lqq
## 1      4  6 1 NA NA 0.6666667 0.3333333 1.333333 0.3333333 4 2 0.5 0.5   3
```

La diferencia es que los tiempos en el sistema (W) han disminuido, ha pasado de 1 a 0.5

```
i_cola3=NewInput.MMC(lambda = 4,mu=3,c=2)
cola3=QueueingModel(i_cola3)
summary(cola3)
```

```
##   lambda mu c  k  m        RO  P0       Lq        Wq X   L   W Wqq Lqq
## 1      4  3 2 NA NA 0.6666667 0.2 1.066667 0.2666667 4 2.4 0.6 0.5   3
```

6.6 Mantenimiento de Maquinaria.

La empresa "Refrigeración Hermanos Pérez" debe elegir entre dos tipos de sistema para el mantenimiento de sus camiones. Se estima que los camiones llegarán al puesto de mantenimiento de acuerdo con una distribución negativa exponencial de media 40 minutos y se cree que este ratio de llegada es independiente del sistema que haya. El primer tipo de sistema puede atender a dos camiones en paralelo, y cada camión se le haría todo el servicio en una media de 30 minutos (el tiempo sigue una distribución exponencial). En el segundo sistema sólo se podría atender a un camión pero el tiempo medio en que se realiza el mantenimiento de un camión es de 15 minutos (distribución exponencial). Para ayudar al encargado de la decisión responda las siguientes cuestiones: a) ¿cuántos camiones habrá por término medio habrá en cualquiera de los dos sistemas? b) ¿Cuánto tiempo pasará cada camión en el taller en cualquiera de los dos sistemas? c) El encargado estima que cada minuto que un camión pasa en el taller reduce los beneficios en 2 euros. Se sabe que el sistema de dos camiones en paralelo tiene un coste de un euro por minuto. ¿Qué debería costar el segundo sistema para que no haya diferencia económica entre los dos?

DATOS. MMC con c=2. lambda=1/40 mu=1/30

```
i_mmc9<-NewInput.MMC(lambda=1/40,mu=1/30,c=2)
q_mmc9<-QueueingModel(i_mmc9)
summary(q_mmc9)
```

```
##   lambda         mu c  k  m    RO        PO        Lq       Wq     X         L
## 1  0.025 0.03333333 2 NA NA 0.375 0.4545455 0.1227273 4.909091 0.025 0.8727273
##          W Wqq Lqq
## 1 34.90909  24 1.6
```

a) Por término medio habrá 0,8727 camiones/hora en el sistema (L).
b) Pasan en el sistema 34,90 minutos de media (W).

Wq: tiempo en cola. Wqq: tiempo en cola cuando hay cola.

c) Van a entrar los camiones que me marque lambda.

```
i_mm10<-NewInput.MM1(lambda=1/40,mu=1/15)
q_mm10<-QueueingModel(i_mm10)
summary(q_mm10)
```

```
##   lambda         mu c  k  m    RO    PO    Lq Wq     X   L  W Wqq Lqq
## 1  0.025 0.06666667 1 NA NA 0.375 0.625 0.225  9 0.025 0.6 24  24 1.6
```

```
Report(q_mm10)
```

```
## The inputs of the M/M/1 model are:
## lambda: 0.025, mu: 0.0666666666666667, n: 0
##
## The outputs of the M/M/1 model are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.625
## The traffic intensity is: 0.375
## The server use is: 0.375
## The mean number of clients in the system is: 0.6
## The mean number of clients in the queue is: 0.225
## The mean number of clients in the server is: 0.375
## The mean time spend in the system is: 24
## The mean time spend in the queue is: 9
## The mean time spend in the server is: 15
## The mean time spend in the queue when there is queue is: 24
## The throughput is: 0.025
```

```r
lambda=1/40
L=q_mm10$L
W=q_mm10$W
W*lambda
```

```
## [1] 0.6
```

M/M/1/K

6.8 Lavadero de Coches.

Un pequeño autoservicio de lavado en el que el coche que entra no puede hacerlo hasta que el otro haya salido completamente, tiene una capacidad de aparcamiento de 10 coches, incluyendo el que está siendo lavado. La empresa ha estimado que los coches llegan siguiendo una distribución de Poisson con una media de 20 coches/hora, el tiempo de servicio sigue una distribución exponencial de 12 minutos. La empresa abre durante 10 horas al día. ¿Cuál es la media de coches perdidos cada día debido a las limitaciones de espacio?

capacidad k=10. lambda=20 coches/hora. mu=60/12=5 coches/hora.

```r
i_mmc6_8<-NewInput.MM1K(lambda = 20,mu=5,k=10)
q_mmc6_8<-QueueingModel(i_mmc6_8)
summary(q_mmc6_8)
```

```
##   lambda mu c  k  m        RO           P0       Lq       Wq        X        L
## 1     20  5 1 10 NA 0.9999993 7.152559e-07 8.66667 1.733335 4.999996 9.666669
##          W     Wqq      Lqq
## 1 1.933335 1.73334 8.666701
```

```r
Report(q_mmc6_8)
```

```
## The inputs of the model M/M/1/K are:
## lambda: 20, mu: 5, k: 10
##
## The outputs of the model M/M/1/K are:
##
## The probability (p0, p1, ..., pk) of the clients in the system are:
## 7.152559e-07 2.861024e-06 1.144409e-05 4.577638e-05 0.0001831055 0.000732422 0.002929688 0.01171875 (
## The probability (q0, q1, ..., qk-1) that a client that enters meets n clients in the system are:
## 2.861026e-06 1.14441e-05 4.577641e-05 0.0001831056 0.0007324226 0.00292969 0.01171876 0.04687504 0.1{
## The traffic intensity is: 0.999999284744092
## The server use is: 0.999999284744092
## The mean number of clients in the system is: 9.66666928927166
## The mean number of clients in the queue is: 8.66667000452757
## The mean number of clients in the server is: 0.999999284744092
## The mean time spend in the system is: 1.93333524068378
## The mean time spend in the queue is: 1.73333524068378
## The mean time spend in the server is: 0.2
## The mean time spend in the queue when there is queue is: 1.7333401998146
## The throughput is: 4.99999642372046
```

Abre 10 horas al día

Llegan 20, y se atienden a 5, se van 15 coches en una hora, luego en 10 horas se irán 150 coches.

Se puede ver así también:

The probability (p0, p1, ..., pk) of the clients in the system are: 7.152559e-07 2.861024e-06 1.144409e-05 4.577638e-05 0.0001831055 0.000732422 0.002929688 0.01171875 0.04687501 0.1875 0.7500002.

k es el número de coches que están en el sistema: uno siendo servido , y puede haber hasta 9 en la cola. La probabilidad de que llegue y esté lleno (pk) es del 75%. 0,75*20 (los que llegan en una hora)= 5 atendidos y 15 no atendidos por hora,o lo que es lo mismo, 150 no atendidos por hora.

M/M/C/K

6.9 Dimensionando el Puerto.

La compañía "Gasolinas y Aceites SA" trabaja con petróleo, como no podía ser de otra manera, que descarga del puerto y lleva a la refinería. En el puerto tiene 6 muelles de descarga y 4 equipos para la descarga del barco. Cuando los muelles están llenos, los barcos se desvía a muelles de espera hasta que les toca su turno. Los barcos llegan según una media de uno cada 2 horas. Para descargar el barco se necesita una media de 10 horas, siguiendo una distribución exponencial. La compañía desea saber los siguientes datos:

a) Por término medio, ¿cuántos barcos hay en el puerto?
b) Por término medio, ¿cuánto tiempo pasa un barco en el puerto?
c) ¿cuál es la media de llegada de los barcos a los muelles de espera?
d) La compañía estudia la posibilidad de construir otro muelle de descarga. La construcción y mantenimiento del puerto costaría X € al año. La compañía estima que desviar un barco hacia los muelles de espera cuando los muelles de descarga están llenos tiene un coste de Y €. ¿Cuál es la relación entre X e Y para que la compañía construya otro puerto de descarga?

lambda=12 barcos al día (llegan cada 2 horas) mu=2,4 (1 cada diez horas)*4 barcos atiende al día

```r
i_puerto<-NewInput.MM1K(lambda = 12,mu=2.4*4,k=6)
q_mmcpuerto<-QueueingModel(i_puerto)
Report(q_mmcpuerto)
```

```
## The inputs of the model M/M/1/K are:
## lambda: 12, mu: 9.6, k: 6
##
## The outputs of the model M/M/1/K are:
##
## The probability (p0, p1, ..., pk) of the clients in the system are:
## 0.06634165 0.08292707 0.1036588 0.1295735 0.1619669 0.2024587 0.2530733
## The probability (q0, q1, ..., qk-1) that a client that enters meets n clients in the system are:
## 0.0888195 0.1110244 0.1387805 0.1734756 0.2168445 0.2710556
## The traffic intensity is: 0.93365834696555
## The server use is: 0.93365834696555
## The mean number of clients in the system is: 3.85756628496461
## The mean number of clients in the queue is: 2.92390793799906
## The mean number of clients in the server is: 0.93365834696555
## The mean time spend in the system is: 0.430382079972244
## The mean time spend in the queue is: 0.326215413305577
## The mean time spend in the server is: 0.104166666666667
## The mean time spend in the queue when there is queue is: 0.358014040932889
## The throughput is: 8.96312013086928
```

```r
i_puerto2<-NewInput.MMCK(lambda = 12,mu=2.4*4,c=4,k=6)
q_mmcpuerto2<-QueueingModel(i_puerto2)
```

```
## Warning in formals(fun): argument is not a function

## Warning in formals(fun): argument is not a function

## Warning in formals(fun): argument is not a function
```

```r
Report(q_mmcpuerto2)
```

```
## The inputs of the model M/M/c/K are:
## lambda: 12, mu: 9.6, c: 4, k: 6
##
## The outputs of the model M/M/c/K are:
##
## The probability (p0, p1, ..., pk) of the clients in the system are:
## 0.2856964 0.3571205 0.2232003 0.09300012 0.02906254 0.009082043 0.002838139
## The traffic intensity is: 1.24645232681043
## The server use is: 0.311613081702607
## The mean number of clients in the system is: 1.26121064727905
## The mean number of clients in the queue is: 0.0147583204686189
## The mean number of clients in the server is: 1.24645232681043
## The mean time spend in the system is: 0.105400027153664
## The mean time spend in the queue is: 0.00123336048699781
## The mean time spend in the server is: 0.104166666666667
## The mean time spend in the queue when there is queue is: 0.0322420634920632
## The throughput is: 11.9659423373801
```

```r
i_puerto3<-NewInput.MMCK(lambda = 12,mu=2.4*4,c=4,k=7)
q_mmcpuerto3<-QueueingModel(i_puerto3)
```

```
## Warning in formals(fun): argument is not a function

## Warning in formals(fun): argument is not a function

## Warning in formals(fun): argument is not a function
```

```r
Report(q_mmcpuerto3)
```

```
## The inputs of the model M/M/c/K are:
## lambda: 12, mu: 9.6, c: 4, k: 7
##
## The outputs of the model M/M/c/K are:
##
## The probability (p0, p1, ..., pk) of the clients in the system are:
## 0.2854432 0.356804 0.2230025 0.09291771 0.02903679 0.009073995 0.002835624 0.0008861324
## The traffic intensity is: 1.24889233453703
## The server use is: 0.312223083634256
## The mean number of clients in the system is: 1.26629597429128
## The mean number of clients in the queue is: 0.0174036397542585
```

```
## The mean number of clients in the server is: 1.24889233453703
## The mean time spend in the system is: 0.105618256280066
## The mean time spend in the queue is: 0.00145158961339982
## The mean time spend in the server is: 0.104166666666667
## The mean time spend in the queue when there is queue is: 0.0354195521698985
## The throughput is: 11.9893664115554
```

```
CompareQueueingModels(q_mmcpuerto2,q_mmcpuerto3)
```

```
##   lambda  mu c k  m        RO        P0         Lq         Wq        X        L
## 1     12 9.6 4 6 NA 0.3116131 0.2856964 0.01475832 0.00123336 11.96594 1.261211
## 2     12 9.6 4 7 NA 0.3122231 0.2854432 0.01740364 0.00145159 11.98937 1.266296
##           W        Wqq      Lqq
## 1 0.1054000 0.03224206 1.238095
## 2 0.1056183 0.03541955 1.360111
```

Pasa más tiempo en el puerto, se puede poner una más en la cola

6.10.

Tenemos una tasa de fallo Calculamos P[>=5]. Queremos que haya pocas máquinas en reparación.

6.12. Mantenimiento de dispensadores

Las máquinas que dispensan billetes para el metro en la compañía "RNFV"se suelen estropear cada 45 horas. Se supone que el único reparador de la estación tarda 4 horas en reparar la máquina. Se asume que los dos tiempos son la media de una distribución exponencial. ¿Cuál es el mínimo número de máquinas que debe haber para asegurarse que haya al menos 5 máquinas en servicio con una probabilidad mayor que 0.95?

Datos: Se estropean cada 45 horas: 1/45 /h mu=1/4 /h

```
library(queueing)
#Unidad de tiempo: dias
lamda= 24/45
mu=6

prob<-numeric(2)
for (i in 5:7) {
  m1<-NewInput.MM1K(lambda = lamda,mu=mu,k=i)
  CheckInput(m1)
  c1<-QueueingModel(m1)
  prob[i-4]=sum(c1$Pn[1:(i-4)])

}
prob
```

```
## [1] 0.9111116 0.9920988 0.9992977
```

```
#Para k=6 se cumpla
```

6.14. Dispensario Gratuito.

Uno de los hospitales de la ciudad de Valencia ofrece todos los miércoles por la noches revisiones gratis de vista. Un test necesita, por término medio, 20 minutos distribuyéndose según una exponencial. Los clientes llegan según una distribución de Poisson de media 6/hora, y los pacientes se atienden según norma FIFO.

Los encargados del hospital desean saber que cantidad de personal sanitario deben disponer. Para ello habría que calcular para diferentes cantidades de doctores:1) ¿cuál es el número medio de gente esperando? 2) el tiempo medio que un cliente pasa en la clínica y 3) el tiempo medio que los doctores están parados.

lambda= 6 /h mu= 3 /h

```
i_mm16.19=NewInput.MM1(lambda = 6,mu=3)
# CheckInput(i_mm6.19) rho es mayor que 1
```

```
#i_mmc6.19=NewInput.MMC(lambda = 6,mu=3,c=2)
#CheckInput(i_mmc6.19) error

i_mmc6.19=NewInput.MMC(lambda = 6,mu=3,c=3)
CheckInput(i_mmc6.19)
q_mmc6.19=QueueingModel(i_mmc6.19)
```

Con c=3 no nos da error el modelo.

a) Número medio de gente esperando

```
cat("El nº medio de clientes en cola es ",q_mmc6.19$L)
```

```
## El nº medio de clientes en cola es  2.888889
```

b)Tiempo medio que un cliente pasa en la clínica

```
summary(q_mmc6.19)
```

```
##   lambda mu c  k  m        RO        P0        Lq        Wq X        L
## 1      6  3 3 NA NA 0.6666667 0.1111111 0.8888889 0.1481481 6 2.888889
##           W       Wqq Lqq
## 1 0.4814815 0.3333333   3
```

```
Report(q_mmc6.19)
```

```
## The inputs of the model M/M/c are:
## lambda: 6, mu: 3, c: 3, n: 0, method: Exact
##
## The outputs of the model M/M/c are:
##
## The probability (p0, p1, ..., pn) of the n = 0 clients in the system are:
## 0.1111111
## The traffic intensity is: 2
## The server use is: 0.666666666666667
## The mean number of clients in the system is: 2.88888888888889
## The mean number of clients in the queue is: 0.888888888888889
## The mean number of clients in the server is: 2
## The mean time spend in the system is: 0.481481481481481
## The mean time spend in the queue is: 0.148148148148148
## The mean time spend in the server is: 0.333333333333333
## The mean time spend in the queue when there is queue is: 0.333333333333333
## The throughput is: 6
```

```
cat("El tiempo medio en el sistema es ",q_mmc6.19$W)
```

```
## El tiempo medio en el sistema es  0.4814815
```

c)Tiempo que los doctores están parados

```
cat("La probabilidad de que no haya nadie es ", q_mmc6.19$Pn)
```

```
## La probabilidad de que no haya nadie es  0.1111111
```

REDES

Red abierta de Jackson

6.19. Centralita Telefónica.

La compañía de seguros "La Otra Vida" tiene una centralita telefónica. Las llamadas llegan según una distribución de media 35 cada hora. Los clientes llaman para dos cosas: para reclamaciones o para solicitar información, para ello deben apretar el botón 1 o el 2. Se cree que el tiempo que tarda un cliente en tomar la decisión y apretar el botón tiene una media de tiempo de 30 segundos según una distribución exponencial. Las llamadas realizadas solo pueden ser procesadas por este contestador de una en una, si alguien llama mientras tanto se le pone una bonita música, se le dice que espere y se le pone en cola. Aproximadamente el 55% de las llamadas son para reclamaciones, el resto para demanda de servicios. El nodo de reclamaciones tiene 3 servidores en paralelo y se estima que el tiempo medio en que atiende un cliente es de 6 minutos (exponencial). El nodo de solicitud de información tiene 7 servidores en paralelo con un tiempo de servicio medio de 20 minutos (exponencial). Se asume que puede haber todos los clientes que se quieran esperando en los nodos. Alrededor del 2% de llamadas que van al nodo de reclamaciones acaban en el de demanda de información, y el 1% que llama al nodo de demanda de información se va al nodo de reclamaciones. Se desea saber por término medio el tamaño de las colas en cada nodo y el tiempo medio que un cliente pasa en el sistema.

lambda= 35 llamadas/h. Pulsar el botón 1 o 2 = 30 seg = 3600/30 /h. Las llamadas son procesadas de 1 en 1. 55% reclamaciones (3 nodos), con mu_r=60/6 clientes/h ; 45% información (7 nodos), con mu_i=3 clientes/h. El 2% de r vana a i, y el 1% de i a r.

Construimos la matriz de transiciones:

```
##           C      R     I
##C          0     0.55  0.45
##R          0      0    0.02
##I          0     0.01   0

datos<-c(0,0.55,0.45,0,0,0.02,0,0.01,0)
probs<-matrix(data = datos, byrow = T,nrow = 3,ncol = 3)
probs
```

```
##      [,1] [,2] [,3]
## [1,]    0 0.55 0.45
## [2,]    0 0.00 0.02
## [3,]    0 0.01 0.00
```

Construimos los modelos

```r
C<- NewInput.MM1(lambda=35,mu=120)
R<- NewInput.MMC(lambda=0,mu=10,c=3)
I<- NewInput.MMC(lambda=0,mu=3,c=7)
CheckInput(C)
CheckInput(R)
CheckInput(I)
q_C=QueueingModel(C)
q_R=QueueingModel(R)
q_I=QueueingModel(I)
```

```r
Central<-NewInput.OJN(prob = probs, C,R,I)
CheckInput(Central)
```

```r
q_Central=QueueingModel(Central)
summary(q_Central)
```

```
##             L         W  X        Lk         Wk       Xk       ROk
## Net 9.898628 0.2828179 35        NA         NA       NA        NA
## Nd1       NA        NA NA 0.4117647 0.01176471 35.00000 0.2916667
## Nd2       NA        NA NA 2.7059261 0.07731217 19.41138 0.6470461
## Nd3       NA        NA NA 6.7809371 0.19374106 16.13823 0.7684870
```

Tiempo medio que un cliente pasa en el sistema

```r
cat("Un cliente pasa de media ",q_Central$W," horas en la centralita")
```

```
## Un cliente pasa de media  0.2828179  horas en la centralita
```

Tamaño de las colas en cada nodo

```r
q_Central$Wk
```

```
## [1] 0.01176471 0.07731217 0.19374106
```

```r
cat("Un cliente espera de media", q_Central$Wk[1], "en el contestador\n", q_Central$Wk[2],
    "si tiene una reclamación y ", q_Central$Wk[3], " si solicita información (en horas)")
```

```
## Un cliente espera de media 0.01176471 en el contestador
##  0.07731217 si tiene una reclamación y  0.1937411  si solicita información (en horas)
```

Si pudiéramos atender a dos llamadas a la vez

```r
C2<- NewInput.MMC(lambda=35,mu=120,c=2)
Central2<-NewInput.OJN(prob = probs, C2,R,I)
CheckInput(Central2)
```

```
q_Central2=QueueingModel(Central2)
summary(q_Central2)
```

```
##            L         W  X        Lk          Wk       Xk       ROk
## Net 9.784868 0.2795676 35        NA          NA       NA        NA
## Nd1       NA        NA NA 0.2980044 0.008514412 35.00000 0.1458333
## Nd2       NA        NA NA 2.7059261 0.077312174 19.41138 0.6470461
## Nd3       NA        NA NA 6.7809371 0.193741060 16.13823 0.7684870
```

```
cat("Un cliente pasa de media ",q_Central2$W," horas en la centralita")
```

```
## Un cliente pasa de media  0.2795676  horas en la centralita
```

```
cat("Un cliente espera de media", q_Central2$Wk[1], "en el contestador\n", q_Central2$Wk[2],
    "si tiene una reclamación y ", q_Central2$Wk[3], " si solicita información (en horas)")
```

```
## Un cliente espera de media 0.008514412 en el contestador
##  0.07731217 si tiene una reclamación y  0.1937411  si solicita información (en horas)
```

Se reduce en 0.2 minutos el tiempo medio.

¿Se puede reducir el número de empleados en siniestros o en atención comercial?

```
RC2<-NewInput.MMC(lambda=0,mu=10,c=2)
q_RC2<-QueueingModel(RC2)
DismS<-NewInput.OJN(prob = probs, C,RC2,I)
CheckInput(DismS)
```

Se puede disminuir el R

```
IC6<-NewInput.MMC(lambda=0,mu=3,c=6)
q_IC6<-QueueingModel(IC6)
CheckInput(IC6)
DismI<-NewInput.OJN(prob = probs, C,R,IC6)
CheckInput(DismI)
```

También se puede reducir I

Ejercicio de clase

Una compañía de seguros tiene un número de atención al cliente al que llegan. las llamadas que llegan según una Poisson, con una media de 35 por hora. La persona que llama tiene una de dos opciones: presionar 1 para comunicar siniestros y presionar 2 para gestiones comerciales. Se estima que la escucha, decisión, y el tiempo de pulsación de un botón es exponencial con una media de 30 segundos. Se puede procesar una llamada a la vez, de modo que mientras una llamada está en proceso, cualquiera de las llamadas que llegan se ponen en cola con buena música de fondo y un agradable mensaje grabado diciendo que por favor espere (asumimos que todos lo hacen). Aproximadamente el 55% de las llamadas van a siniestros,y el resto al servicio de pólizas. El nodo de procesamiento de siniestros tiene tres servidores paralelos, y se estima que los tiempos de servicio son exponenciales con media 6 min. El nodo de gestión comercual tiene siete servidores paralelos, de nuevo con tiempos de servicio exponenciales, y aquí el tiempo medio de servicio es de 20 min. Todos los búferes frente a los nodos se puede suponer que retienen tantas llamadas como entran en las colas. Alrededor del 2% de los clientes que llaman a siniestrsos luego pasan al de gestión de pólizas servicio, y alrededor del 1 % de los clientes que llaman al servicio de pólizas se mandan a siniestros. Se desea conocer los tamaños de cola promedio en cada nodo y el tiempo promedio que un cliente pasa en el sistema.

DATOS:

lambda= 35 llamadas/h. Pulsar el botón 1 o 2 = 30 seg = 3600/30 /h. Las llamadas son procesadas de 1 en 1. 55% siniestros (3 nodos), con mu_r=60/6 clientes/h ; 45% a pólizas (7 nodos), con mu_i=3 clientes/h. El 2% de siiestros van a a pól., y el 1% al revés.

Construimos la matriz de transiciones:

```
##              C        S     P
##C             0       0.55  0.45
##S             0        0    0.02
##P             0       0.01   0

datos<-c(0,0.55,0.45,0,0,0.02,0,0.01,0)
probs<-matrix(data = datos, byrow = T,nrow = 3,ncol = 3)
probs
```

```
##      [,1] [,2] [,3]
## [1,]    0 0.55 0.45
## [2,]    0 0.00 0.02
## [3,]    0 0.01 0.00
```

Construimos los modelos

```
C<- NewInput.MM1(lambda=35,mu=120)
S<- NewInput.MMC(lambda=0,mu=10,c=3)
P<- NewInput.MMC(lambda=0,mu=3,c=7)
CheckInput(C)
CheckInput(S)
CheckInput(P)
q_C=QueueingModel(C)
q_S=QueueingModel(S)
q_P=QueueingModel(P)
```

```
Compañia<-NewInput.OJN(prob = probs, C,S,P)
CheckInput(Compañia)
```

```
q_Compañia=QueueingModel(Compañia)
summary(q_Compañia)
```

```
##             L         W  X        Lk         Wk       Xk      ROk
## Net 9.898628 0.2828179 35        NA         NA       NA       NA
## Nd1       NA        NA NA 0.4117647 0.01176471 35.00000 0.2916667
## Nd2       NA        NA NA 2.7059261 0.07731217 19.41138 0.6470461
## Nd3       NA        NA NA 6.7809371 0.19374106 16.13823 0.7684870
```

Tiempo medio que un cliente pasa en el sistema

```
cat("Un cliente pasa de media ",q_Compañia$W," horas en la compañía")
```

```
## Un cliente pasa de media  0.2828179  horas en la compañía
```

Tamaño de las colas en cada nodo

```
q_Compañia$Wk
```

```
## [1] 0.01176471 0.07731217 0.19374106
```

```
cat("Un cliente espera de media", q_Central$Wk[1], "en el contestador\n", q_Central$Wk[2],
    "si quiere ser atendido en siniestros y ", q_Central$Wk[3], " si solicita el servicio de pólizas (e
```

```
## Un cliente espera de media 0.01176471 en el contestador
##  0.07731217 si quiere ser atendido en siniestros y  0.1937411  si solicita el servicio de pólizas (e
```

6.21 Reparaciones Electrónicas.

Una empresa de reparación de electrónica sirve a la mayoría de los minoristas de electrodomésticos de la región. Recibe aparatos para arreglar según una distribución de Poisson de media 9 a la hora. Todos los aparatos nada más llegar son inspeccionados por un especialista que determina a que sección debe ir dependiendo del tipo de reparación si es básica, si la debe ver un especialista, si debe enviar el aparato al fabricante y por tanto mandados a un almacén para ser enviado. Alrededor del 17% es enviado a fabrica. De los restantes, el 57% va a reparaciones generales y el 43% es enviado a un experto. Todos los aparatos reparados van al almacén para ser enviados, sin embargo el 5% que va a reparaciones generales vuelve al inicio para ser nuevamente clasificado. Debido a la variedad de los aparatos enviados y la variedad de problemas la distribución exponencial es una adecuada representación para la clasificación, reparación y envío. Solo hay una persona en la selección y tarda una media de 6 minutos por aparato. Hay tres personas en reparaciones generales y tardan por término medio 35 minutos por aparato (incluidos los que son devueltos a clasificación). Hay cuatro expertos y tardan por término medio 65 minutos en reparar un aparato (estos aparato siempre salen arreglados). Hay dos muelles de embarque, cada uno de ellos tarda una media de 12,5 minutos en embalar un aparato. El ingeniero de la empresa se pregunta cuantos aparatos hay por término medio en cada nodo, el tiempo que pasa en cada nodo y el tiempo medio que está un aparato en la empresa desde que es recibido y clasificado hasta que es empaquetado.

Datos: lambda=9 aparatos/h. Una persona con mu=60/6=10 /h. El 17% se envía a fábrica, 57% reparaciones generales (el 5% vuelve al inicio) y 43% expertos. Después son enviados, excepto ese 5%. 3 personas en reparaciones generales con mu=60/35 aparatos/h. 4 expertos con mu=65/60 ap/h. Envío: hay 2 muelles de embarque con mu=60/12.5 ap/h.

Introducimos la matriz de prob de transición entre etapas.

```
     #I     #F      #RG         #E
#I   0     0.17  0.57*0.83   0.43*0.83
#F   0       0       0           0
#RG  0.05   0.95     0           0
#E   0       1       0           0
```

```
datos<-c(0,0.17,0.57*0.83,0.43*0.83,
         0,0,0,0,
         0.05,0.95,0,0,
         0,1,0,0)

probs<-matrix(data = datos,byrow = T,nrow = 4,ncol = 4)
```

Creamos los modelos de las distintas etapas.

```
I<-NewInput.MM1(lambda=9,mu=10)
q_I<-QueueingModel(I)
```

```
RG<-NewInput.MMC(lambda = 0,mu=60/35,c=3)
q_RG<-QueueingModel(RG)
```

```
E<-NewInput.MMC(lambda = 0,mu=60/65,c=4)
q_E<-QueueingModel(E)
```

```
F<-NewInput.MMC(lambda = 0,mu=60/12.5,c=2)
q_F<-QueueingModel(F)
```

Creamos la red.

```
Empresa<-NewInput.OJN(prob = probs, I,F,RG,E)
CheckInput(Empresa)
```

```
q_empresa<-QueueingModel(Empresa)
summary(q_empresa)
```

```
##           L        W  X        Lk        Wk       Xk       ROk
## Net 43.7286 4.858733  9        NA        NA       NA        NA
## Nd1      NA       NA NA 11.788591 1.3098435 9.218053 0.9218053
## Nd2      NA       NA NA 15.483871 1.7204301 9.000000 0.9375000
## Nd3      NA       NA NA  6.599085 0.7332317 4.361061 0.8479841
## Nd4      NA       NA NA  9.857048 1.0952276 3.289923 0.8910208
```

· Cuántos aparatos hay por término medio en cada nodo:

```
Ap=q_empresa$Lk
Ap
```

```
## [1] 11.788591 15.483871  6.599085  9.857048
```

```
cat("De media hay ", Ap[1], " aparatos en el inicio, ", Ap[2], "en la fábrica,","\n",Ap[3], " en repara
```

```
## De media hay  11.78859  aparatos en el inicio,  15.48387 en la fábrica,
##  6.599085  en reparaciones generales y  9.857048  en expertos
```

· El tiempo que pasa en cada nodo:

```
Tiempo=q_empresa$Wk
cat("De media pasa ", Tiempo[1], " horas en el inicio, ", Tiempo[2], "en la fábrica,","\n",Tiempo[3], "
```

```
## De media pasa  1.309843  horas en el inicio,  1.72043 en la fábrica,
##  0.7332317  en reparaciones generales y  1.095228  en expertos
```

· El tiempo medio que está un aparato en la empresa desde que es recibido y clasificado hasta que es empaquetado.

```r
cat("Un aparato pasa en media ",q_empresa$W, " horas en la empresa")
```

```
## Un aparato pasa en media  4.858733  horas en la empresa
```

6.23 Aglomerados JPK.

Una empresa de fabricación de puertas de madera tiene una unidad de negocio que fabrica puertas de muebles de cocina. Dichas puertas, de dimensiones diferentes según pedidos, reciben un tratamiento en 3 etapas. El número de puertas que la unidad de negocio fabrica son alrededor de 50000 puertas al año. La primera etapa es capaz de procesar 220 puertas al día. La segunda etapa consta de dos máquinas que procesan cada una 140 puertas al día. La tercera etapa es una etapa manual, para la que se dispone de 3 trabajadores que tardan aproximadamente 5 minutos por puerta. Los días tienen 480 minutos y los años 240 días. Se pueden suponer tiempos distribuidos según una negativa exponencial tanto para las llegadas de pedidos como para los ritmos de producción.

a) ¿Cuál es el número de puertas que habrá en cada etapa, incluyendo las puertas en las máquinas y las que están siendo procesadas por los operarios?

```r
l1=50000/240
m1=220 #puertas/día

i_mmc1<-NewInput.MM1(lambda = l1,mu=m1)
q_mmc1<-QueueingModel(i_mmc1)
summary(q_mmc1)
```

```
##     lambda  mu c  k  m        RO        PO       Lq         Wq        X
## 1 208.3333 220 1 NA NA 0.9469697 0.0530303 16.91017 0.08116883 208.3333
##          L          W        Wqq      Lqq
## 1 17.85714 0.08571429 0.08571429 18.85714
```

```r
l2=l1
m2=140

i_mmc2<-NewInput.MMC(lambda = l2,mu=m2,c=2,n=10)
q_mmc2<-QueueingModel(i_mmc2)
summary(q_mmc2)
```

```
##     lambda  mu c  k  m        RO        PO       Lq          Wq        X
## 1 208.3333 140 2 NA NA 0.7440476 0.1467577 1.845503 0.008858413 208.3333
##          L          W        Wqq      Lqq
## 1 3.333598 0.01600127 0.01395349 3.906977
```

```r
l3=l1
m3=480/5

i_mmc3<-NewInput.MMC(lambda = l3,mu=m3,c=3)
q_mmc3<-QueueingModel(i_mmc3)
summary(q_mmc3)
```

```
##     lambda mu c  k  m        RO         PO       Lq          Wq        X
## 1 208.3333 96 3 NA NA 0.7233796 0.08559656 1.378369 0.006616173 208.3333
##          L          W       Wqq      Lqq
## 1 3.548508 0.01703284 0.0125523 3.615063
```

Con Redes (considerando que tenemos que ir obligatoriamente de la Etapa 1 a la 2, y de ahí a la 3, nos encontramos con la siguiente matriz):

```
#     E1  E2  E3
# E1  0   1   0
# E2  0   0   1
# E3  0   0   0


Prob=matrix(data=c(0,1,0,0,0,1,0,0,0),byrow = T,3,3)

red1=NewInput.MM1(lambda = l1,mu=m1)
red2=NewInput.MMC(lambda = 0,mu=m2,c=2)
red3=NewInput.MMC(lambda = 0,mu=m3,c=3)

red <- NewInput.OJN(prob=Prob, red1, red2, red3)
q_red=QueueingModel(red)
summary(q_red)
```

```
##              L         W        X        Lk         Wk       Xk       ROk
## Net 24.73925 0.1187484 208.3333        NA         NA       NA        NA
## Nd1       NA        NA       NA 17.857143 0.08571429 208.3333 0.9469697
## Nd2       NA        NA       NA  3.333598 0.01600127 208.3333 0.7440476
## Nd3       NA        NA       NA  3.548508 0.01703284 208.3333 0.7233796
```

b) ¿En que afectaría al sistema anterior que en la etapa segunda se colocara un limitador de capacidad, mediante el cual no se aceptaran al almacén previo a dicha etapa más de 5 puertas?

En primer lugar, el sistema dejaría de ser una serie de colas convencional porque se limita la capacidad de una de ellas. Esta sería una cola M/M2/7. Pero para saber cómo afectaría, lo mejor es saber el porcentaje de veces que el almacén estaría lleno (P(n=7)=4,1%).

```
cat("La probabilidad de que haya 6 personas en la cola es ", q_mmc2$Pn[7])
```

```
## La probabilidad de que haya 6 personas en la cola es  0.04980067
```

Por tanto durante un 3,8% de las ocasiones la primera etapa no podría trabajar al estar bloqueado el sistema posterior, dado que la primera etapa trabaja al 95%, la probabilidad de que el bloqueo del sistema afecte a la producción total es alta.

c) Si en el sistema original el tiempo medio de entrega de una puerta es de 5 días. ¿Cuántas puertas hay?

El número de puertas en el sistema es L=lambda · W=1041'7 puertas

d) Suponga que en el sistema original la demanda de puertas asciende a 70000 puertas/año. Si se opta por no comprar una máquina nueva en la primera etapa, ¿Cuántas horas extra al día debe trabajar la primera máquina? ¿En qué afectaría dicho cambio a los plazos de entrega? ¿Cómo se comportarían los almacenes?

```
l11=70000/240 #puertas/día
l11
```

```
## [1] 291.6667
```

```
#Tengo que fabricar al menos 291,77-220=71,67 puertas más al día
#Como trabaja 8 h al día, la capacidad de producción es
#220/8 puertas por hora, que son  27.5
#71.67/27.6=algo más de 2.6

m11=300
m11
```

```
## [1] 300
```

Deben trabajar al menos 2.6 horas más al día

```
i_mmc11<-NewInput.MM1(lambda = l11,mu=m11)
q_mmc11<-QueueingModel(i_mmc11)
summary(q_mmc11)
```

```
##     lambda  mu c  k  m        RO         P0       Lq         Wq        X  L    W
## 1 291.6667 300 1 NA NA 0.9722222 0.02777778 34.02778 0.1166667 291.6667 35 0.12
##    Wqq Lqq
## 1 0.12  36
```

```
l22=l11
m2=140/8

i_mmc2<-NewInput.MMC(lambda = l2,mu=m2,c=2)
# q_mmc2<-QueueingModel(i_mmc2) rho es mayor que 1
summary(q_mmc2)
```

```
##     lambda  mu c  k  m        RO        P0       Lq          Wq        X
## 1 208.3333 140 2 NA NA 0.7440476 0.1467577 1.845503 0.008858413 208.3333
##          L          W        Wqq      Lqq
## 1 3.333598 0.01600127 0.01395349 3.906977
```

```
l3=l1
m3=480/5

i_mmc3<-NewInput.MMC(lambda = l3,mu=m3,c=3)
q_mmc3<-QueueingModel(i_mmc3)
summary(q_mmc3)
```

```
##     lambda mu c  k  m        RO         P0       Lq          Wq        X
## 1 208.3333 96 3 NA NA 0.7233796 0.08559656 1.378369 0.006616173 208.3333
##          L          W       Wqq      Lqq
## 1 3.548508 0.01703284 0.0125523 3.615063
```

Suponga que en la segunda etapa no hay dos si no tres máquinas. Dichas máquinas tardan en estropearse 3 días desde que se arreglan y un mecánico tarda de media 5 horas en arreglarlas cada vez. Sólo se dispone de un mecánico. ¿Tiene este sistema suficiente capacidad para hacer frente a la demanda?

```
#Suponemos que cada 8 horas llega una máquina (ese es nuestro día, la jornada)
lambda=1
mu=8/5

i_mm1kk<-NewInput.MM1KK(lambda = 1,mu=8/5,k=3)
q_mm1kk=QueueingModel(i_mm1kk)
summary(q_mm1kk)
```

```
##   lambda  mu c k  m        RO        P0        Lq        Wq        X        L
## 1      1 1.6 1 3 NA 0.8503799 0.1496201 0.7890123 0.5798969 1.360608 1.639392
##          W       Wqq      Lqq
## 1 1.204897 0.8653846 1.384615
```

El tiempo en cola es medio día, y el tiempo en servicio es 1,2 de media (W).

```
i_mm1km<-NewInput.MMCKM(lambda = 1,mu=8/5,k=3,m=3,c=1)
q_mm1km=QueueingModel(i_mm1km)
summary(q_mm1km)
```

```
##   lambda  mu c k m        RO        P0        Lq        Wq        X        L
## 1      1 1.6 1 3 3 0.8503799 0.1496201 0.7890123 0.5798969 1.360608 1.639392
##          W       Wqq      Lqq
## 1 1.204897 0.8653846 1.384615
```

Habrá momentos en los que no pueda atender (rho=0,85),con esas tasas de servicio habrá problemas. (modelizamos como un modelo mm1kk)

Este es un sistema con fuente finita. El número de máquinas que estarán siendo reparadas por término medio es de 0,67. Por tanto se tienen 2,33 máquinas trabajando, y por tanto es más que suficiente ya que 2,33*140>208.

f) (1 punto) Sobre el caso anterior ¿Qué porcentaje de tiempo sólo hay una máquina trabajando? ¿Y ninguna? ¿Qué ocurre con los almacenes durante este tiempo que hay menos de dos máquinas trabajando?

Sólo hay una máquina trabajando cuando hay dos estropeadas P2=13,4% No hay ninguna trabajando si todas están estropeadas P3=2,7% Cuando hay menos de dos máquinas trabajando el nivel de almacén antes de la segunda etapa crece. Y por tanto aumenta el plazo total de entrega.

g) (2 puntos) Sobre el caso anterior ¿Qué opinión le merece que vaya uno de los trabajadores de la tercera sección a ayudar al mecánico cuando haya dos o más máquinas estropeadas? Debe sustentar la opinión con datos, suponga para ello que el trabajador de la tercera sección se comporta como un mecánico más, cuando trabaja como tal.

A priori, si hay dos máquinas estropeadas, el ritmo al cual pasan puertas a la tercera etapa es de 140 puertas/día. Con dos operarios es posible abastecer 192 puertas al día, con lo que no pasaría nada. Además el porcentaje de veces que hay dos máquinas estropeadas es muy bajo en el nuevo sistema.

h) (1 punto) ¿Cuál sería en el caso anterior la probabilidad de que hubiera más de una máquina estropeada?

La probabilidad de que hubiera más de una máquina estropeada sería 8%

6.25 Juguetes KP.

Una empresa de transformados plásticos tiene 4 secciones (A,B,C,D). Los productos que fabrica se pueden clasificar en 5 categorías, con demandas anuales diferentes. Los productos de categoría 1 tienen una demanda anual de alrededor de 500 unidades, y sus especificaciones los harán circular por la sección A, luego la sección B y por último la sección C. Los productos de categoría 2 tienen una demanda anual de alrededor de 3000 unidades, y sus especificaciones los harán circular por la sección A, luego la sección B y por último la sección D. Los productos de categoría 3 tienen una demanda anual de alrededor de 2000 unidades, y sus especificaciones los harán circular por la sección B y la sección D. Los productos de categoría 4 tienen una demanda anual de alrededor de 2000 unidades, y sus especificaciones los harán circular por la sección A y la sección C. Los productos de categoría 5 tienen una demanda anual de alrededor de 1000 unidades, y sus especificaciones los harán circular por la sección B y la sección C. Sabiendo que el ritmo de producción por hora en una máquina de tipo A es de 2 unidades, el de B es de 2 unidades, el de C es de 4 unidades y el de D es de 2 unidades por hora. Sabiendo que el año tiene 220 días laborables de 8 horas cada uno y asumiendo tiempos de servicio exponenciales.

Preguntas a) Modele el problema definiendo los parámetros básicos para cada sección (lambda, mu). b) Defina el número de máquinas imprescindibles en cada sección. c) Asumiendo que los niveles de inventario se mantendrán en los mínimos imprescindibles, ¿Cuál es el tiempo medio esperado de producción de un producto en el sistema? d) Si el tiempo medio de entrega de un producto es de 10 días laborables ¿Cuál es el nivel medio de inventario en el sistema? e) Suponga que en la sección B hacen falta dos máquinas. La experiencia con esas máquinas indica que requieren un cierto mantenimiento con distribución exponencial con media cada 5 días y que el tiempo que dura dicho mantenimiento se distribuye exponencialmente con media de 1 día. ¿Cuántas máquinas hacen falta para que haya al menos dos máquinas funcionando el 90% del tiempo?. f) En el caso anterior, ¿cuántas máquinas habrá en funcionamiento por término medio?

Tenemos un red de Jackson abierta de 4 nodos. La llegada externa a cualquier nodo es Poisson Yi (sumamos los posibles "caminos").

```
YA=5500
YB=3000
YC=0
YD=0
```

Todos los servidores de cada etapa tiene un servicio exponencial de media mu_i De cada etapa i un cliente se mueve a otra etapa con probabilidad rij, y al exterior con probabilidad ri,0.

```
rAB=3500/YA #Sumamos cuándo podemos ir de A a B entre todas las veces que salimos de A
rAC=2000/YA
rBC=1500/6500
rBD=5000/6500

#Metemos la matriz con las prob de transición

prob<-matrix(c(0,0.64,0.36,0,0,0,0.23,0.77,0,0,0,0,0,0,0,0),byrow=T,4,4)
```

Miro cuántas máquinas necesito viendo la producción necesaria para cubrir la demanda.

```
A<-NewInput.MMC(lambda=5500,mu=3520,n=0,c=2)
B<-NewInput.MMC(lambda=3000,mu=3520,n=0,c=2)
C<-NewInput.MM1(lambda=0,mu=7040,n=0)
```

```r
D<-NewInput.MMC(lambda=0,mu=3520,n=0,c=2)

fabrica<-NewInput.OJN(prob = prob,A,B,C,D)
q_fabrica<-QueueingModel(fabrica)
summary(q_fabrica)
```

```
##              L          W    X         Lk           Wk     Xk       ROk
## Net 20.90873 0.00245985 8500         NA           NA     NA        NA
## Nd1       NA         NA   NA  4.0100251 0.0004717677 5500.0 0.7812500
## Nd2       NA         NA   NA 13.0192875 0.0015316809 6520.0 0.9261364
## Nd3       NA         NA   NA  0.9773059 0.0001149772 3479.6 0.4942614
## Nd4       NA         NA   NA  2.9021102 0.0003414247 5020.4 0.7131250
```

Tenemos dinero para poner una máquina más, la pondríamos en C, que tiene menor congestión (medimos la congestión con ROK, y cuanto más cercano a 1, mayor congestión).

c) Mínimos imprescindibles. Tiempo medio esperado de producción

```r
cat(q_fabrica$W, "años, o lo que es lo mismo, ",q_fabrica$W*220*8 , " horas")
```

```
## 0.00245985 años, o lo que es lo mismo,  4.329337  horas
```

d) Si el tiempo medio de entrega de un producto es de 10 días laborables ¿Cuál es el nivel medio de inventario en el sistema?

e) Suponga que en la sección B hacen falta dos máquinas. La experiencia con esas máquinas indica que requieren un cierto mantenimiento con distribución exponencial con media cada 5 días y que el tiempo que dura dicho mantenimiento se distribuye exponencialmente con media de 1 día. ¿Cuántas máquinas hacen falta para que haya al menos dos máquinas funcionando el 90% del tiempo?.

MM1K lambda= 2*1/5 máquinas/día mu=1 máquinas/día

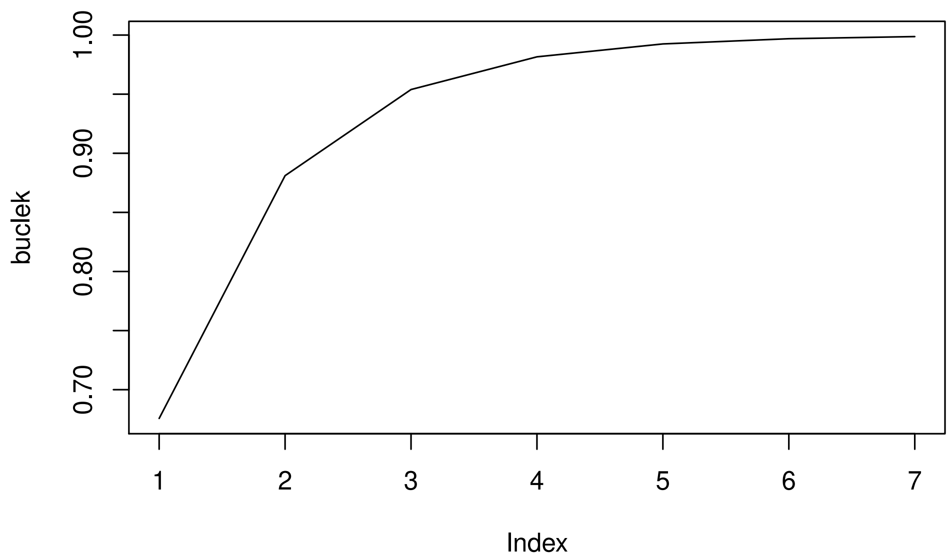
```r
B2<-NewInput.MMCK(lambda = 2/5,mu=1,c=2,k=2)
q_B2<-QueueingModel(B2)
summary(q_B2)
```

```
##   lambda mu c k  m        RO        PO Lq Wq         X         L W Wqq Lqq
## 1    0.4  1 2 2 NA 0.1891892 0.6756757  0  0 0.3783784 0.3783784 1  NA  NA
```

```r
buclek=numeric(7)
for(i in 2:8){
  Maq=NewInput.MMCK(lambda=i/5,mu=1,c=i,k=i)
  q_Maq=QueueingModel(Maq)
  buclek[i-1]=sum(q_Maq$Pn[1:(i-1)])
}
buclek
```

```
## [1] 0.6756757 0.8810573 0.9539237 0.9815951 0.9925034 0.9969051 0.9987096
```

```r
plot(buclek, type = "l")
```



Se verifica para c=4.

f) En el caso anterior, ¿cuántas máquinas habrá en funcionamiento por término medio?

k-numero de máquinas en cola

Casos

7.1 Colas en el parque de atracciones.

Acaba usted de llegar a la cola del aparcamiento de un parque temático. Por lo que le han comentado las colas empiezan con los cajeros a los que paga 6 Euros por coche que entra en el aparcamiento. La segunda cola empieza mientras espera que le asignen una plaza del aparcamiento. La tercera es la cola para pagar la entrada (a 8 euros más por persona), y por último una cola para que comprueben que no pretende entrar comida en el parque temático. Una vez dentro del Parque comenzará una serie de colas que acabará con su paciencia y sus piernas, pero eso será después de haber conseguido traspasar esa última barrera. No todo el mundo que entra en el Parque sigue el mismo camino, pero su interés radica en saber cuánto tardará usted en alcanzar el interior del Parque. Un breve análisis de la situación le indica los siguientes datos. Entran aproximadamente 2400 coches por hora, para ser atendidos por 20 cajas en paralelo que tardan en cobrar aproximadamente 29 segundos por cliente. De todos los coches que entran un 18% van al aparcamiento VIP (al que usted no va ni irá y por eso no sabe cómo funciona). El 82% restante va al aparcamiento convencional que, de un modo muy eficiente es capaz de aparcar los coches, de uno en uno, con un tiempo de ciclo promedio de 3,5 segundos por coche. Mediante otros medios (trenes y autobuses) se acercan junto con los clientes en coche particular nuevos grupos de clientes. Se calcula que en el parque que hemos elegido para depositar (y perder) nuestros ahorros, entran aproximadamente 25.000 personas al día en 6000 grupos (los grupos son importantes porque compran las entradas de modo conjunto). De los 6000 grupos sólo 3500 grupos compran las entradas en taquilla (los otros ya las compraron por agencia o llevan un pase de varios días comprado anteriormente). Todos los clientes llegan aproximadamente en las 3 primeras horas de apertura del parque.

Los que han de pagar tendrán que hacer 6 colas para pagar en 30 cajas (cada cola alimenta a 5 cajas). En cada caja tardarán en promedio 92 segundos en atenderles. Tras pagar queda la última cola donde cada cliente pasa de modo individual, y pasan todos: los que acaban de comprar y los que venían con ticket precomprado, por el detector de comidas y bebidas. Estos son 12 carriles en paralelo, cada uno con su propia cola, que tardan 5 segundos en dejar pasar a cada cliente.

a) Si cada coche mide 4 metros, cuantos metros de carretera hacen falta para que quepan en promedio todos los coches que se pondrán en cola delante de las cajas de aparcamiento.

Primera cola.

Datos: LLegan 2400 coches por hora durante 3 horas. Hay 20 cajas de aparcamiento en paralelo. Distribución del tiempo de servicio: 29 segundos por cliente.

```
library(queueing)
modelo1=NewInput.MMC(lambda = 2400,mu = 3600/29,c = 20)
CheckInput.i_MMC(modelo1)
cola1=QueueingModel.i_MMC(modelo1)
options(digits = 4)
```

```
cat("Longitud media de la cola:", cola1$Lq,"\n")
```

```
## Longitud media de la cola: 24.14
```

```
cat("Metros necesarios: ", 4*cola1$Lq,"\n")
```

```
## Metros necesarios:  96.57
```

b) ¿Cuánto tiempo se tarda en hacer la cola para aparcar el coche, una vez haya pagado el aparcamiento?

Datos: El 82 % de 2400 coches/hora van al aparcamiento ordinario. Tiempo de aparcacoches de 1 en 1: 1 servidor. Distribución del tiempo de servicio de aparcacoches 3.5 segundos por coche. Los que llegan a aparcar son los que ya han pasado por las cabinas para pagar. La llegada es en serie entre estos dos nodos para los que llegan en coches, como estamos en colas en serie de M/M/c/infinito, se tiene que la salida es lo mismo que la entrada, por lo que llegan 2400 al aparcamiento.

```
cat("Llegan ",0.82*2400," coches por hora al parking")
```

```
## Llegan  1968  coches por hora al parking
```

```
modelo2a=NewInput.MM1(lambda = 0.82*2400,mu=3600/3.5)
e <- simpleError("test error")
tryCatch(CheckInput.i_MM1(modelo2a), error = function(e) e, finally = cat("\n"))
```

```
## Throughput is 1028.57142857143
## Utilization is 191.333333333333%
```

```
## <simpleError in CheckInput.i_MM1(modelo2a): ro is greater or equal to one!!>
```

Da error, con un solo servidor no es posible.

Son 2400 por h, y llegan en las 3 primeras horas.

Es un modelo infactible con c=1, ya que la cola crecería indefinidamente. En este caso, al estimarse un máximo de 5904=0.8224003 coches, se necesitarían de media 5.74 (0.822400 3*3.5/3600) horas para aparcarlos. Para aplicar los modelo de colas rho debe ser menor que 1, lo que se consigue si el número de servidores (aparcacoches) es mayor que 1.91.

```
0.82*2400/(3600/3.5)
```

```
## [1] 1.913
```

Asumimos entonces que c>=2.

```
modelo2=NewInput.MMC(lambda = 0.82*2400,mu=3600/3.5,c=2)
CheckInput.i_MMC(modelo2)
cola2=QueueingModel.i_MMC(modelo2)
cat("Tiempo medio en la cola 2 es de ",cola2$Wq*3600, " segundos")
```

```
## Tiempo medio en la cola 2 es de  37.78  segundos
```

c) ¿Cuánto tiempo tardaremos en conseguir nuestra entrada desde que hemos aparcado el coche, considerando que tarda 5 minutos desde que aparca hasta que llega a la cola?

Datos: 6000 grupos de los que sólo 3500 compran en taquilla. 6 colas, 30 cajas, cada cola (carril) tiene 5 cajas. 6 colas iguales, independientes. 3500 personas al día, por hora es 3500/3 y entre 6 colas.

Ya hemos aparcado. Analizamos una cola, por eso dividimos entre 3*6 (3 horas y 6 colas). Cada cola tiene c=5 cajas.

```
modelo3=NewInput.MMC(lambda = 3500/(3*6),mu = 3600/92,c=5)
CheckInput.i_MMC(modelo3)
cola3=QueueingModel.i_MMC(modelo3)
cat("El tiempo medio hasta que conseguimos las entradas",(cola3$W *60)+5," minutos")
```

```
## El tiempo medio hasta que conseguimos las entradas 55.45  minutos
```

d) ¿Cuánta gente habrá como usted haciendo cola para pagar?

```
cat("Número medio de personas haciendo cola: ",cola3$Lq, "en cada cola MM5, \n En total",6*cola3$Lq)
```

```
## Número medio de personas haciendo cola:  158.5 en cada cola MM5,
##  En total 951.1
```

e) ¿Cuánto tiempo tardaremos en entrar en el parque desde que aparcamos?.

Al nodo de seguridad llegan 25000 personas en las 3 horas (lambda=2500/3) Hay 12 carriles en paralelo, cada uno con su cola lo que supone tener 12 colas independientes. Distribución del servicio: 5 segundos por persona (3600/5 por hora).

```r
modelo4=NewInput.MM1(lambda = 25000/(12*3),mu=3600/5)
CheckInput.i_MM1(modelo4)
cola4=QueueingModel.i_MM1(modelo4)
summary(cola4)
```

```
##   lambda  mu c  k  m     RO      P0    Lq      Wq     X     L       W     Wqq
## 1  694.4 720 1 NA NA 0.9645 0.03549 26.21 0.03774 694.4 27.17 0.03913 0.03913
##     Lqq
## 1 28.17
```

```r
cat("Tiempo total en el tercer nodo: ",cola3$W*60, " minutos y en el cuarto ",60*cola4$W, "\n en total
```

```
## Tiempo total en el tercer nodo:  50.45  minutos y en el cuarto  2.348
##  en total : 57.79
```

f) ¿qué ocurrirá en el sistema de cajas si el tiempo de atención en la caja de compra de entradas el tiempo medio de atención es de 100 segundos?

```r
modelo3a=NewInput.MMC(lambda = 3500/(3*6),mu = 3600/100,c=5)
tryCatch(CheckInput.i_MMC(modelo3a), error = function(e) e, finally = cat("\n"))
```

```
## Throughput is: 180
## Utilization exceeds 100% use!!: 108.024691358025%
```

```
## <simpleError in CheckInput.i_MMC(modelo3a): ( lambda/(mu*c) ) has to be less than one!!>
```

Tenemos que el modelo no es aplicable, habría que aumentar c. En este caso al estimarse un máximo de 3500 personas, se necesitarían de media (3500x100)/(3600x30)=3.24 horas para vender todas las entradas.

```r
3500*100/(3600*30)
```

```
## [1] 3.241
```

g) ¿qué repercusión tendría en la cola posterior dicha alteración?

Como cada persona compra para un grupo, se beneficiarían los que llegan con las entradas compradas, al disminuir las llegadas por hora en la cola 4.

h) Con las condiciones de e, ¿cuál es la repercusión de añadir un carril adicional de venta de entradas?

```r
modelo3b=NewInput.MMC(lambda = 3500/(3*7),mu = 3600/92,c=5)
CheckInput.i_MMC(modelo3b)
cola3b=QueueingModel.i_MMC(modelo3b)
summary(cola3b)
```

```
##   lambda    mu c  k  m     RO       P0    Lq      Wq     X     L       W    Wqq
## 1  166.7 39.13 5 NA NA 0.8519 0.008349 3.785 0.02271 166.7 8.045 0.04827 0.0345
##    Lqq
## 1 6.75
```

```
cat("Disminuiría el tiempo en el nodo de compra de entradas en ", (cola3$W-cola3b$W)*60 , "minutos,\n s
```

```
## Disminuiría el tiempo en el nodo de compra de entradas en  47.55 minutos,
##  siendo el nuevo tiempo en la etapa de compra de entradas de 2.896  minutos
```

```
cat("En total, con los 7 carriles, la persona del enunciado tardaría desde que llega a la primera cola h
```

```
## En total, con los 7 carriles, la persona del enunciado tardaría desde que llega a la primera cola ha
## En total,6 carriles tarda  59.57 minutos
```

7.5 Mantenimiento PECAJU

El servicio técnico de PECAJU tiene 3 etapas relevantes y necesarias para todos los productos que maneja. Cada una de ellas es relativamente manual y la capacidad productiva de la misma es directamente proporcional al número de personas que trabajan en la misma. El número de productos que se reciben en dicho servicio técnico es de 41 unidades al día y la llegada de los mismos sigue un proceso de Poisson.

Tras la tercera etapa hay un proceso de control de calidad que revisa el producto obtenido. Es también una etapa manual y se podría considerar una cuarta etapa. Tras la inspección un cierto porcentaje de productos son devueltos a la etapa 1, otros a la etapa 2 y otros a la etapa 3. Por la configuración del producto, una vez un producto vuelve a la etapa 1 debe seguir el proceso preestablecido hasta el final.

Los datos de cada etapa están en la tabla adjunta. Los tiempos de operación en cada etapa están expresado en horas y se ajustan razonablemente bien a una distribución exponencial.

Los productos que se fabrican son específicos para cada cliente. El cliente tiene una cierta urgencia en recoger su producto acabado y por ello el tiempo de espera del mismo es un tema relevante. Los datos de cada una de las etapas es el siguiente:

f. Diseñe el sistema que con un mínimo número de personas total cumple con los requerimientos.

g. Defina los parámetros básicos del sistema: número de pedidos promedios, tiempo de espera medio, numero medio de pedidos por etapa.

Número de pedidos promedio: vuelven un 10% de los pedidos al sistema, (5,3,y 2), luego de media habrá 1,1*41 pedidos

Salen del sistema los mismos elementos que entran, no hay destrucción, excepto el primer día, que empezamos desde 0. Eventualmente, como la tasa nos permite atender a la cola que se crea, saldrán 41, los mismos que entran.

Los datos de tiempo de espera parecen muy elevados. Se le ocurren varias maneras de atacar el problema.

h. Contratar una persona más. Si tuviera que proponer la contratación de una persona más ¿dónde la pondría y por qué? ¿qué efecto tendrá sobre el sistema?

i. Invertir en alguna de las diferentes etapas para reducir a la mitad la tasa de fallos. ¿en cuál y por qué? ¿qué efecto tendría en el sistema?

j. Invertir en alguna de las diferentes etapas para reducir a la mitad la variabilidad del proceso. ¿en cuál y por qué? ¿qué efecto tendría?

Hay 4 etapas: calculemos la matriz de probabilidad.

```
##      Et1   Et2   Et3   Et4
## Et1   0     1     0     0
## Et2   0     0     1     0
## Et3   0     0     0     1
## Et4 0.02 0.03  0.05    0      (las que llegan desde el control de calidad)


Prob=matrix(data=c(0,1,0,0,0,0,1,0,0,0,0,1,0.02,0.03,0.05,0),byrow = T,4,4)
dim(Prob)
```

```
## [1] 4 4
```

Tenemos que hacer 41/ci*mui<1.

```
l1=41/8
m1=1/3

library(queueing)
```

```
41/8/(16*m1)<1
```

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

```
41/8/(15*m1)<1
```

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

```
red1=NewInput.MMC(lambda = l1,mu=m1, c=16)
```

```
m2=1/4
41/8/(21*m2)<1
```

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

```
41/8/(20*m2)<1
```

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

```
red2=NewInput.MMC(lambda = 0,mu=m2,c=22)
```

```
m3=1/2
41/8/(12*m3)<1
```

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

```
41/8/(13*m3)<1
```

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

```r
red3=NewInput.MMC(lambda = 0,mu=m3,c=12)
```

```r
m4=1

41/8/(6*m4)<1
```

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

```r
red4=NewInput.MMC(lambda = 0,mu=m4,c=6)
```

```r
red <- NewInput.OJN(prob=Prob, red1, red2, red3, red4)
q_red=QueueingModel(red)
summary(q_red)
```

```
##         L     W     X    Lk     Wk    Xk    ROk
## Net 191.1 37.29 5.125    NA     NA    NA     NA
## Nd1    NA    NA    NA 66.67 13.009 5.239 0.9823
## Nd2    NA    NA    NA 76.23 14.875 5.410 0.9836
## Nd3    NA    NA    NA 26.40  5.152 5.694 0.9491
## Nd4    NA    NA    NA 21.78  4.250 5.694 0.9491
```