

案例：意见领袖与智能穿戴设备扩散

吴翔

2019-10-20

概述

我们通过案例来阐述如何使用社会网络分析研究健康管理。所有分析过程均通过R语言实现。

本案例源自市场营销中的常用场景：

新产品在初期扩散较为缓慢的时候，是否可能借助意见领袖的口碑来促进产品的扩散？

接下来，我们将通过论文导读与R语言实现两个部分了解中心性在市场营销中的具体应用。

问题背景

2012年谷歌眼镜的亮相，被称作“智能穿戴设备元年”。在智能手机的创新空间逐步收窄和市场增量接近饱和的情况下，智能穿戴设备作为智能终端产业下一个热点已被市场广泛认同。

智能穿戴设备可以用作：

- 慢性病管理
- 生活与运动习惯监测

而今智能手环、智能手表等产品已经非常丰富，然而大多数产品的销售状况并不理想。在市场营销中，为了促进新产品的扩散，企业可能采用“product seeding program”，亦即：

选取意见领袖作为“种子客户”，免费向其提供新产品，以期这些“种子客户”能够通过口碑效应，促进新产品的扩散。

然而，社会网络中的哪些客户是“意见领袖”呢？

我们求诸节点中心性。

研究思路

对于任意行动者 i ，购买智能穿戴设备的决策受到两个途径的影响：

- 大众传播：例如广告等途径，其特点是具有全局效果（可以近似认为广告覆盖了所有目标人群）
- 口碑传播：通过个体之间传播，其特点是具有局部效果（每个人只受其家人朋友的影响）

因此，如果第 $t - 1$ 期末行动者 i 尚未购买智能穿戴设备，那么他在第 t 期购买智能穿戴设备的概率是：

$$\text{prob}_{it} = 1 - (1 - p) \times (1 - q)^{m_{it}},$$

其中：

- p ：创新系数，用以刻画大众传播
- q ：模仿系数，用以刻画口碑传播
- m_{it} ：在第 $t - 1$ 期末，与行动者 i 直接相连的行动者中，已经购买了智能穿戴设备的数量

如果行动者 i 在第 t 期购买了智能穿戴设备，则记作 $y_{it} = 1$ ；否则，记作 $y_{it} = 0$ 。所以，每期的销售量为：

$$\text{sales}_t = \sum_i y_{it}.$$

进一步，考虑到货币的时间价值，可以通过折现率 r 来计算企业通过销售智能穿戴设备获得的收益净现值：

$$\text{npv} = \sum_t \frac{\text{sales}_t}{(1+r)^t}.$$

正常产品扩散情形

我们首先搭建仿真模型，以模拟该新型智能穿戴设备在投放市场之后的扩散过程。

创建并初始化社会网络

```
rm(list = ls())
suppressMessages(library(igraph))
suppressMessages(library(ggplot2))
suppressMessages(library(dplyr))
suppressMessages(library(tidyr))
set.seed(123)

# create a graph
n <- 5000
g <- random.graph.game(n = n, p = 0.002)

# set parameters
setattr <- function(g, seeds = NULL) {
  # set vertex attributes
  V(g)$adopted <- 0
  if (length(seeds) > 0) {
    # seeds
    V(g)$adopted[seeds] <- 1
  }
  # set graph attributes
  graph_attr(g, "adopters") <- list(sum(V(g)$adopte
d))
  # return the graph
  return(g)
}
```

新产品扩散机制

智能穿戴设备的扩散过程中：

- 行动者之间是相互沟通和影响的
- 影响模式由以上公式刻画

```

iteration <- function(g, p = 0.1, q = 0.01) {
  # Number of vertices
  N <- vcount(g)
  # find non-adopters
  non_adopters <- as.vector(V(g)[V(g)$adopted == 0])
  # get the adjacency matrix
  adjacencymat <- as_adjacency_matrix(g, sparse = TRUE)
  # extract relevant adjacency matrix
  na_adjacency <- adjacencymat[non_adopters, , drop = FALSE]
  # remove the large matrix object
  rm(adjacencymat)
  # initialize the number of adopters
  nums_adopted <- unlist(g$adopters)
  # initialize the number of periods in which continuous zero adoption occurs
  num_test <- 0
  # iterate over all none-adopted vertices
  while (length(non_adopters) != 0 && num_test < 10) {
    # adopted judge
    # number of i's acquaintances who have adopted the good by time t
    mit <- as.vector(na_adjacency %*% V(g)$adopted)
    pit <- 1 - (1 - p) * (1 - q) ^ mit
    flag <- runif(length(non_adopters)) < pit
    # update the adopted status
    flagid <- non_adopters[which(flag == TRUE)]
    if (length(flagid)) {

```

```
V(g)[flagid]$adopted <- TRUE
}
# update the non-adopters id
nonflagid <- which(flag == FALSE)
non_adopters <- non_adopters[flag == FALSE]
# update the relevant adjacency matrix
na_adjacency <- na_adjacency[nonflagid, , drop
= FALSE]
# update the number of new adopters
nums_adopted <- c(nums_adopted, sum(flag))
# update the number of periods in which continu
ous zero adoption occurs
if (sum(flag) == 0) {
  num_test <- num_test + 1
}else {
  num_test <- 0
}
}
# store as a graph attribute
graph_attr(g, "adopters") <- list(nums_adopted)
# return the graph
return(g)
}
```

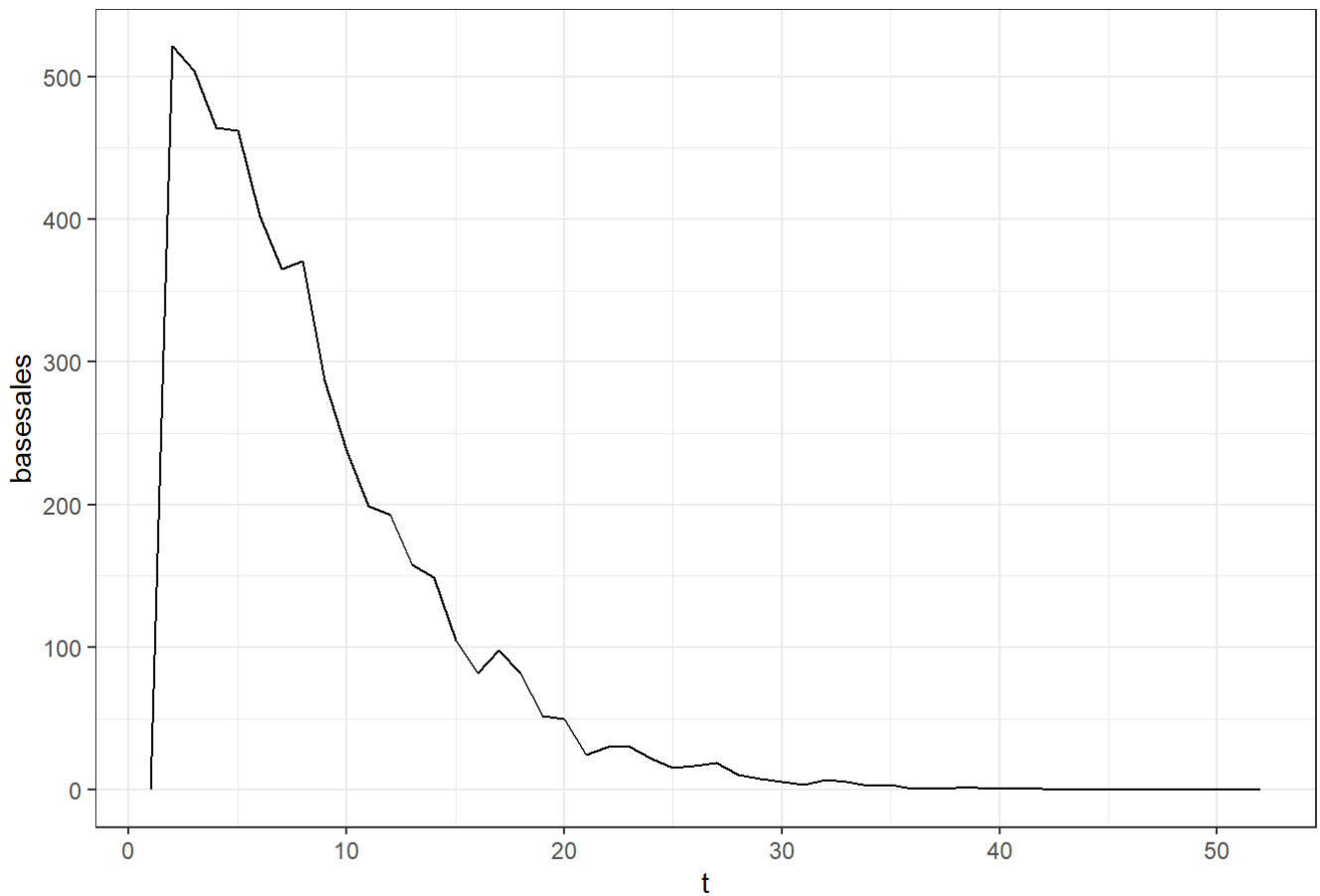
根据新产品扩散过程，我们可以计算对应的净现值（net present value, NPV）。

```
# create a function to calculate the NPV
npv <- function(sales, discount.rate = 0.05) {
  # calculate npv
  npv <- sum(sales / (1 + discount.rate) ^ (1:length(sales)))
  return(npv)
}
```

新产品销售量

我们根据以上仿真模型，计算该智能穿戴设备的销售量。

```
# base model
# set attributes
g.base <- setattr(g)
# diffusion process
g.base <- iteration(g.base)
# obtain the sales
sales.base <- graph_attr(g.base, "adopters")[[1]]
sales.base <- data.frame(t = 1:length(sales.base), base
sales = sales.base)
# plot the sales
ggplot(sales.base, aes(x = t, y = basesales)) + geom_line() + labs(caption = "Sales over periods for the new wearable device") + theme_bw()
```



Sales over periods for the new wearable device

同时可以看到，总销售量为4999。

净现值

考虑到货币的时间价值（2017年的1000元比2018年的1000元人民币更值钱：通货膨胀），我们计算销售额的净现值。

```
# print the results
npv.base <- npv(sales.base$basesales)
npvvec <- data.frame(npv.base = npv.base)
npvvec
```

```
##      npv.base
## 1      3443
```

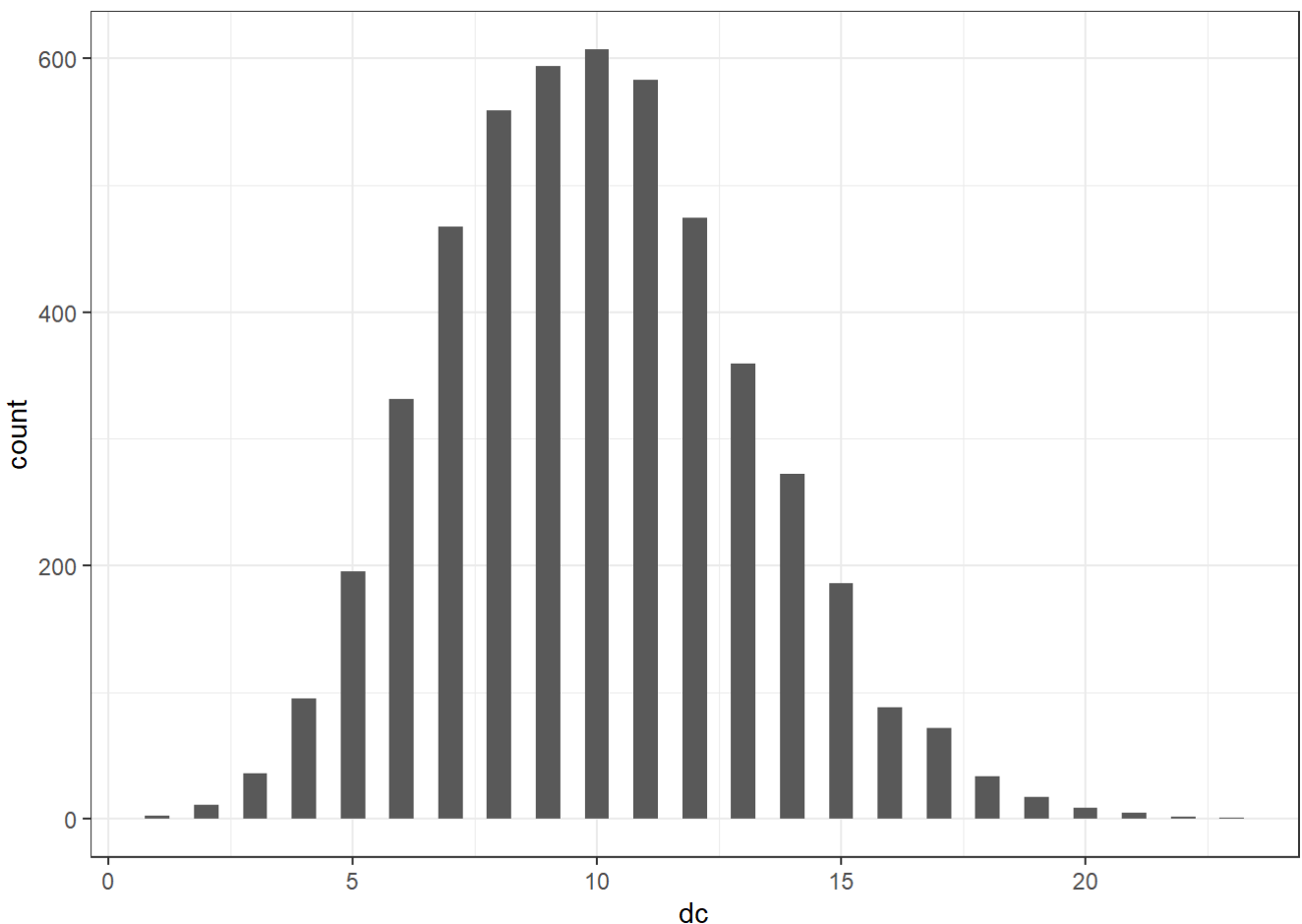
营销策略及结果

我们依次考虑四种营销策略。

度中心性

首先，我们计算节点的度中心性，并绘制度中心性的分布图。

```
# calculate degree centrality
dc <- degree(g)
# distribution of dc
data.frame(dc = dc) %>% ggplot(aes(x = dc)) + geom_histogram(binwidth = 0.5) + theme_bw()
```



进而，我们选取10个度中心性最大的行动者（即消费者）作为“种子客户”，免费向其提供智能穿戴产品。

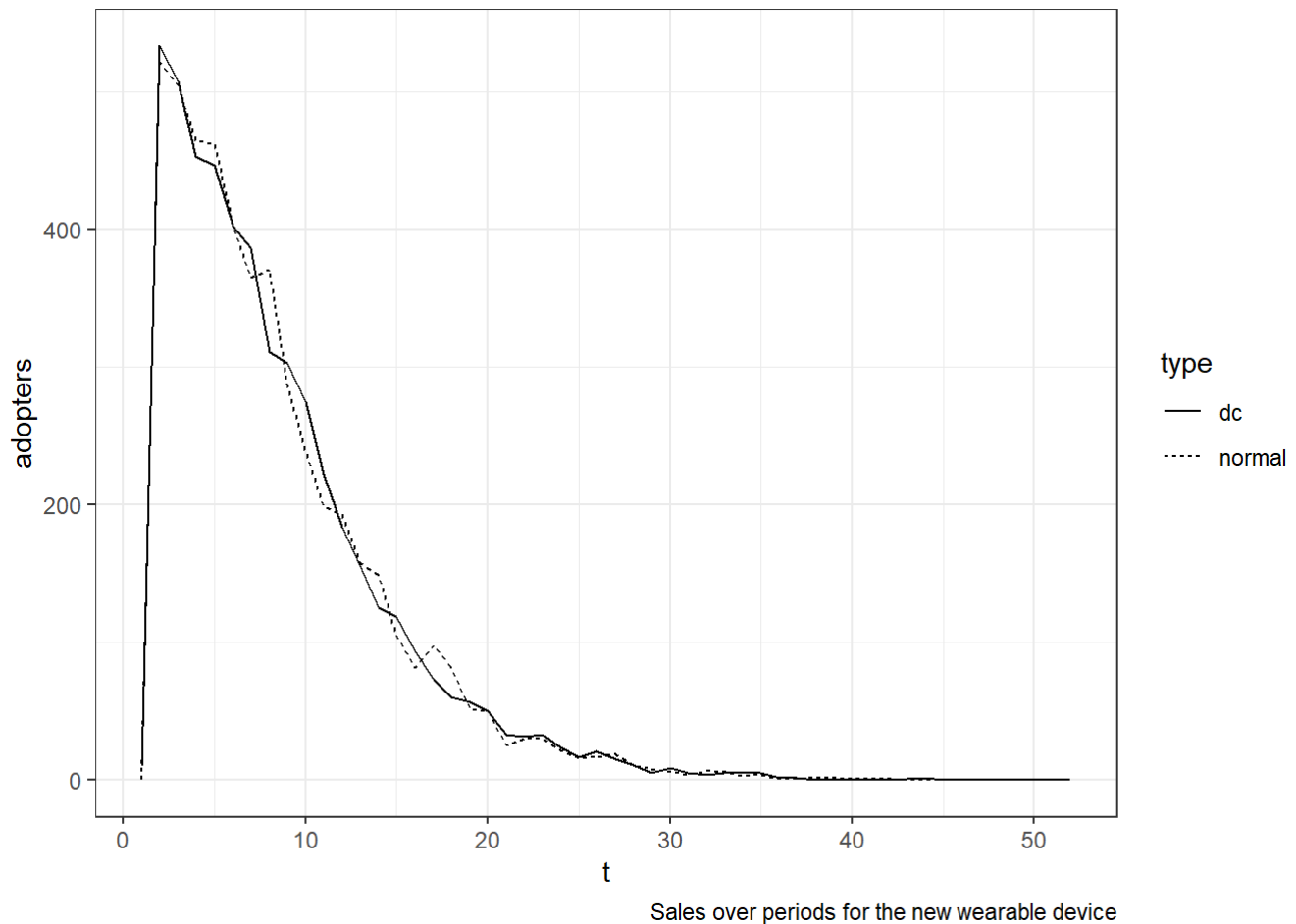
```
# select 2/1000 actors as seeds
seeds_size <- n * 0.002
seeds_dc <- order(dc, decreasing = T)[1:seeds_size]
# print the dc of these seeds
dc[seeds_dc]
```

```
## [1] 23 22 22 21 21 21 21 21 20 20
```

可以看到，以上“种子客户”的度（未归一化）都大约在20左右。

```
# set attributes
g.dc <- setattr(g, seeds = seeds.dc)
# diffusion process
g.dc <- iteration(g.dc)
# obtain the sales
sales.dc <- graph_attr(g.dc, "adopters")[[1]]

# create a data frame
t <- 1:max(length(sales.base$basesales), length(sales.d
c))
# adding zeros
addzeros <- function(x, tmax) {
  if (length(x) < tmax) {
    # adding zeros
    x <- c(x, rep(0, tmax - length(x)))
  }
  # return
  return(x)
}
# apply for two vectors
illust.dat <- data.frame(t, normal = addzeros(sales.bas
e$basesales, length(t)), dc = addzeros(sales.dc, length
(t)))
illust.dat <- gather(illust.dat, "type", "adopters", 2:
3)
# plot the results
ggplot(illust.dat, aes(x = t, y = adopters, linetype =
type)) + geom_line() + labs(caption = "Sales over peri
ods for the new wearable device") + theme_bw()
```



可以看到，总销售量为5000。

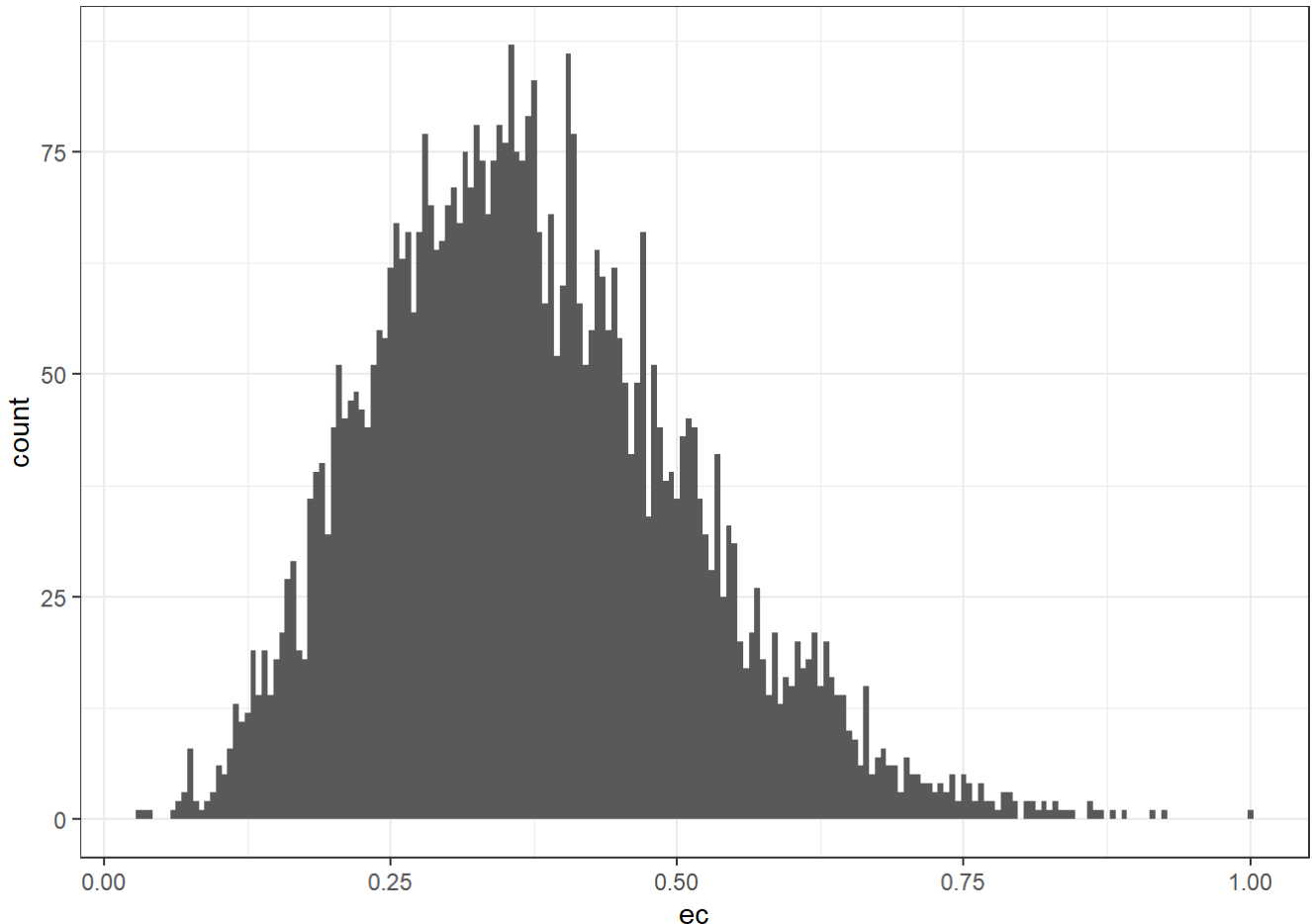
```
# calculate and compare NPV
npv.dc <- npv(sales.dc)
npvvec <- cbind(npvvec, npv.dc)
npvvec
```

```
##      npv.base npv.dc
## 1      3443    3446
```

特征向量中心性

首先，我们计算节点的特征向量中心性，并绘制特征向量中心性的分布图。

```
# calculate eigenvector centrality
ec <- eigen_centrality(g, scale = T)$vector
# distribution of ec
data.frame(ec = ec) %>% ggplot(aes(x = ec)) + geom_histogram(binwidth = 5e-3) + theme_bw()
```



进而，我们选取10个特征向量中心性最大的行动者（即消费者）作为“种子客户”，免费向其提供智能穿戴产品。

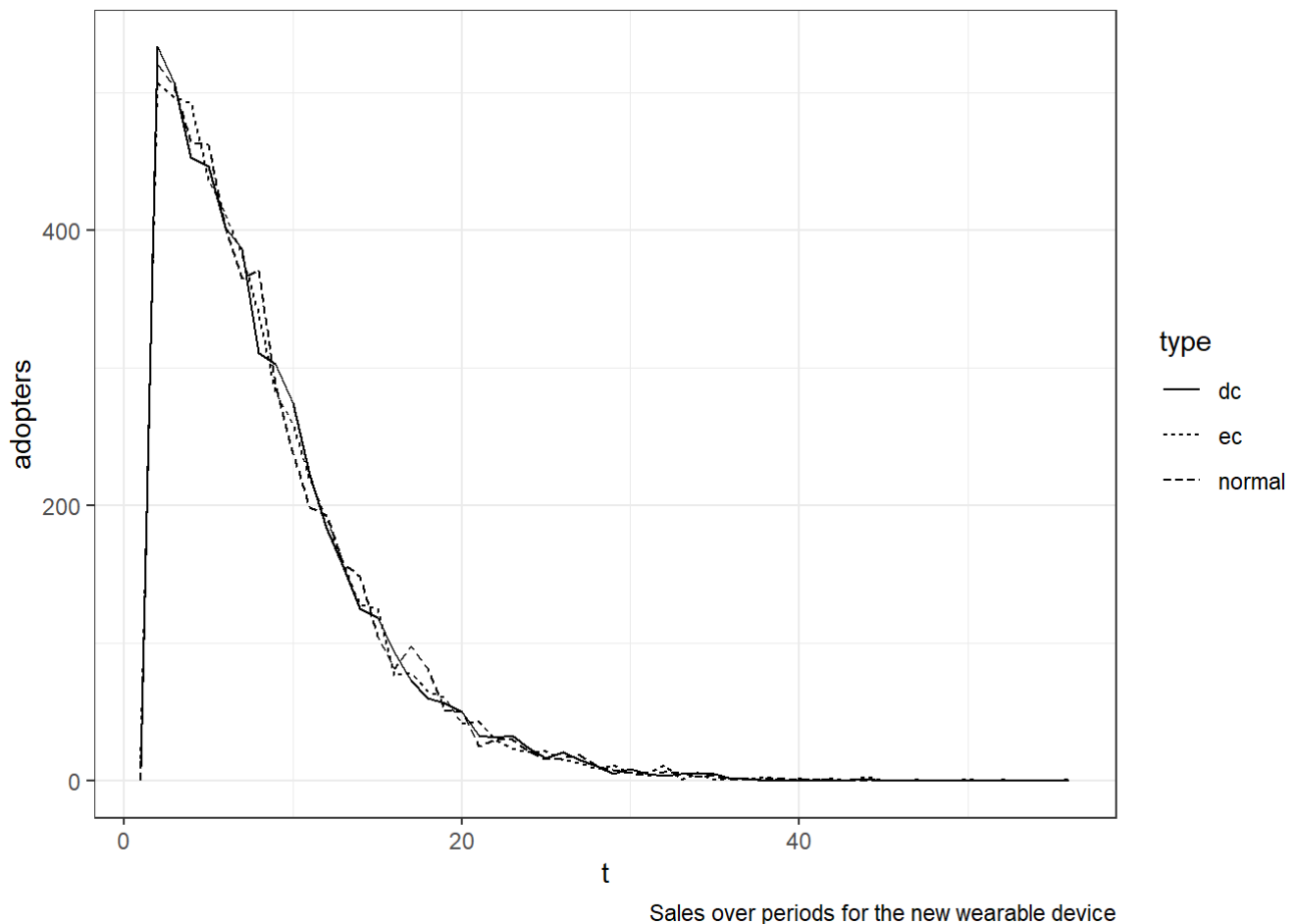
```
# select 2/1000 actors as seeds
seeds.ec <- order(ec, decreasing = T)[1:seedsizesize]
# print the cc of these seeds
ec[seeds.ec]
```

```
## [1] 1.000 0.926 0.915 0.889 0.878 0.872 0.866 0.862
0.861 0.845
```

```
# set attributes
g.ec <- setattr(g, seeds = seeds.ec)
# diffusion process
g.ec <- iteration(g.ec)
# obtain the sales
sales.ec <- graph_attr(g.ec, "adopters")[[1]]

# create a data frame
t <- 1:max(length(sales.base$basesales), length(sales.dc), length(sales.ec))
# adding zeros
addzeros <- function(x, tmax) {
  if (length(x) < tmax) {
    # adding zeros
    x <- c(x, rep(0, tmax - length(x)))
  }
  # return
  return(x)
}
# apply for two vectors
illust.dat <- data.frame(t,
                        normal = addzeros(sales.base$basesales, length(t)),
                        dc = addzeros(sales.dc, length(t)),
                        ec = addzeros(sales.ec, length(t)))
illust.dat <- gather(illust.dat, "type", "adopters", 2:4)
# plot the results
```

```
ggplot(illust.dat, aes(x = t, y = adopters, linetype =
  type)) + geom_line() + labs(caption = "Sales over peri
ods for the new wearable device") + theme_bw()
```



可以看到，总销售量为5000。

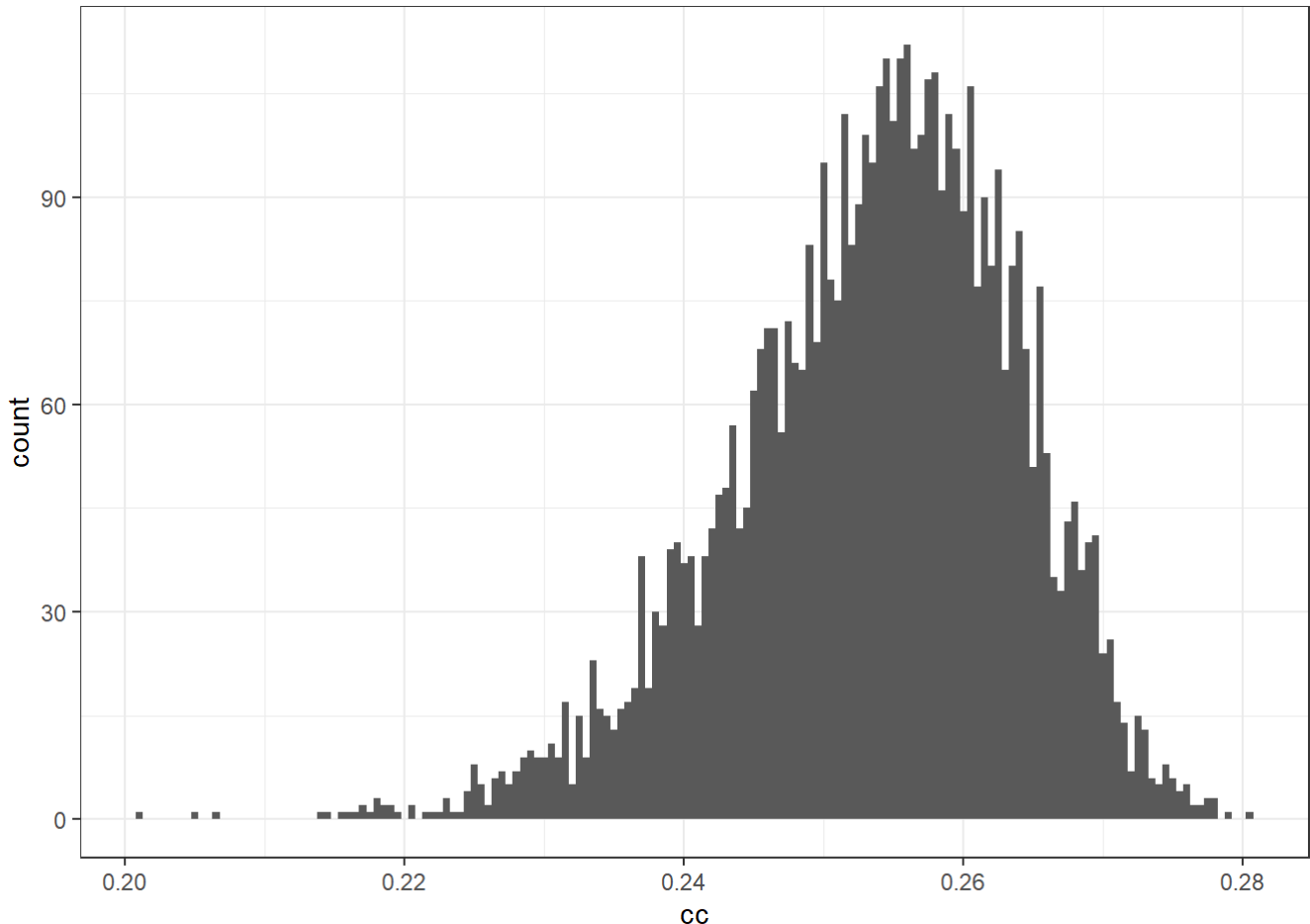
```
# calculate and compare NPV
npv.ec <- npv(sales.ec)
npvvec <- cbind(npvvec, npv.ec)
npvvec
```

```
##    npv.base npv.dc npv.ec
## 1      3443   3446   3442
```

接近中心性

首先，我们计算节点的接近中心性，并绘制接近中心性的分布图。

```
# calculate closeness centrality
cc <- closeness(g, normalized = T)
# distribution of cc
data.frame(cc = cc) %>% ggplot(aes(x = cc)) + geom_histogram(binwidth = 5e-4) + theme_bw()
```



进而，我们选取10个接近中心性最大的行动者（即消费者）作为“种子客户”，免费向其提供智能穿戴产品。

```
# select 2/1000 actors as seeds
seeds.cc <- order(cc, decreasing = T)[1:seedsizesize]
# print the cc of these seeds
cc[seeds.cc]
```

```
## [1] 0.280 0.279 0.278 0.278 0.278 0.278 0.277 0.277
0.277 0.277
```

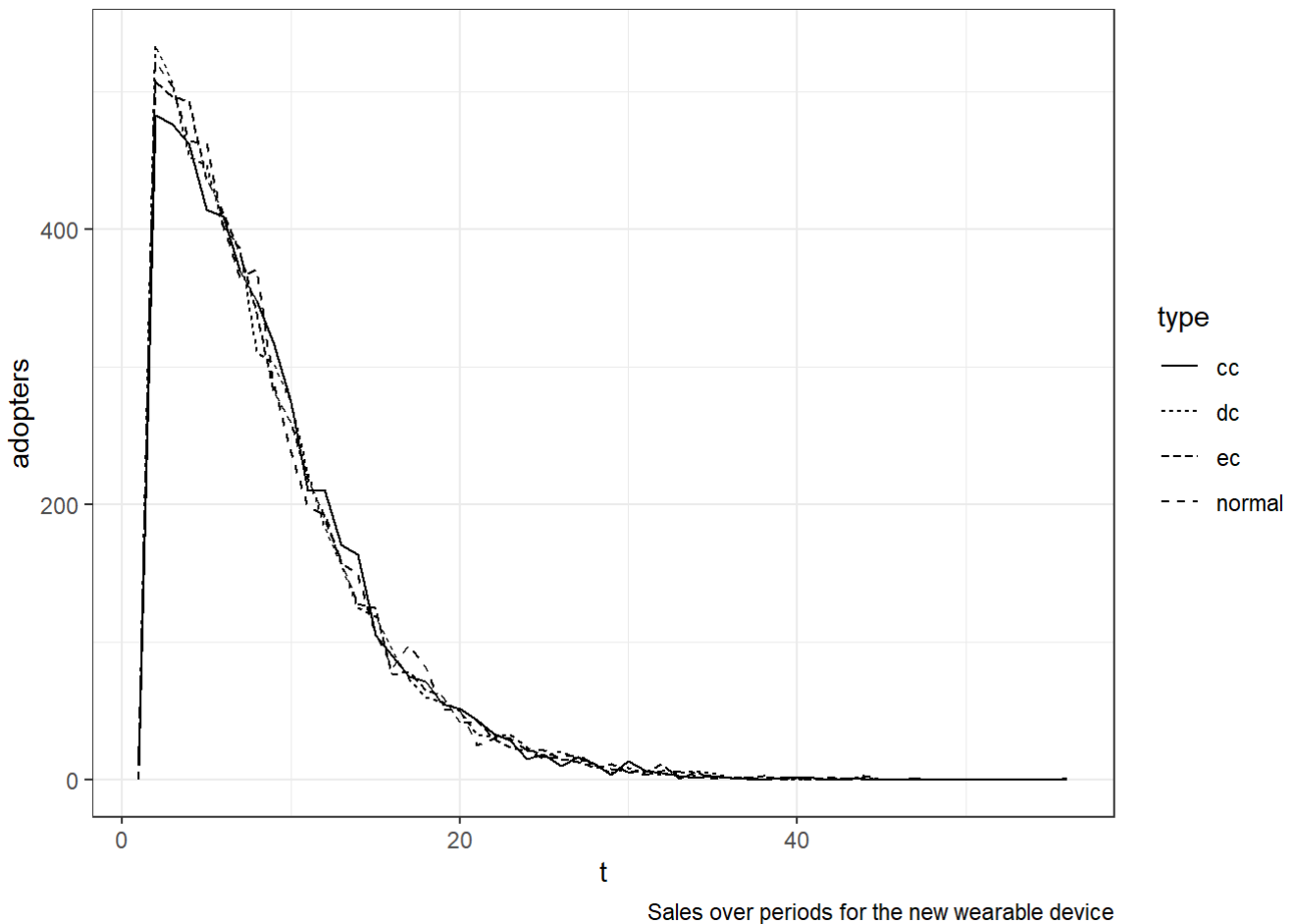


```
# set attributes
g.cc <- setattr(g, seeds = seeds.cc)
# diffusion process
g.cc <- iteration(g.cc)
# obtain the sales
sales.cc <- graph_attr(g.cc, "adopters")[[1]]

# create a data frame
t <- 1:max(length(sales.base$basesales), length(sales.dc), length(sales.ec), length(sales.cc))
# adding zeros
addzeros <- function(x, tmax) {
  if (length(x) < tmax) {
    # adding zeros
    x <- c(x, rep(0, tmax - length(x)))
  }
  # return
  return(x)
}
# apply for two vectors
illust.dat <- data.frame(t,
                          normal = addzeros(sales.base$basesales, length(t)),
                          dc = addzeros(sales.dc, length(t)),
                          ec = addzeros(sales.ec, length(t)),
                          cc = addzeros(sales.cc, length(t)))
illust.dat <- gather(illust.dat, "type", "adopters", 2:
```

5)

```
# plot the results
ggplot(illust.dat, aes(x = t, y = adopters, linetype =
  type)) + geom_line() + labs(caption = "Sales over peri
ods for the new wearable device") + theme_bw()
```



可以看到，总销售量为4999。

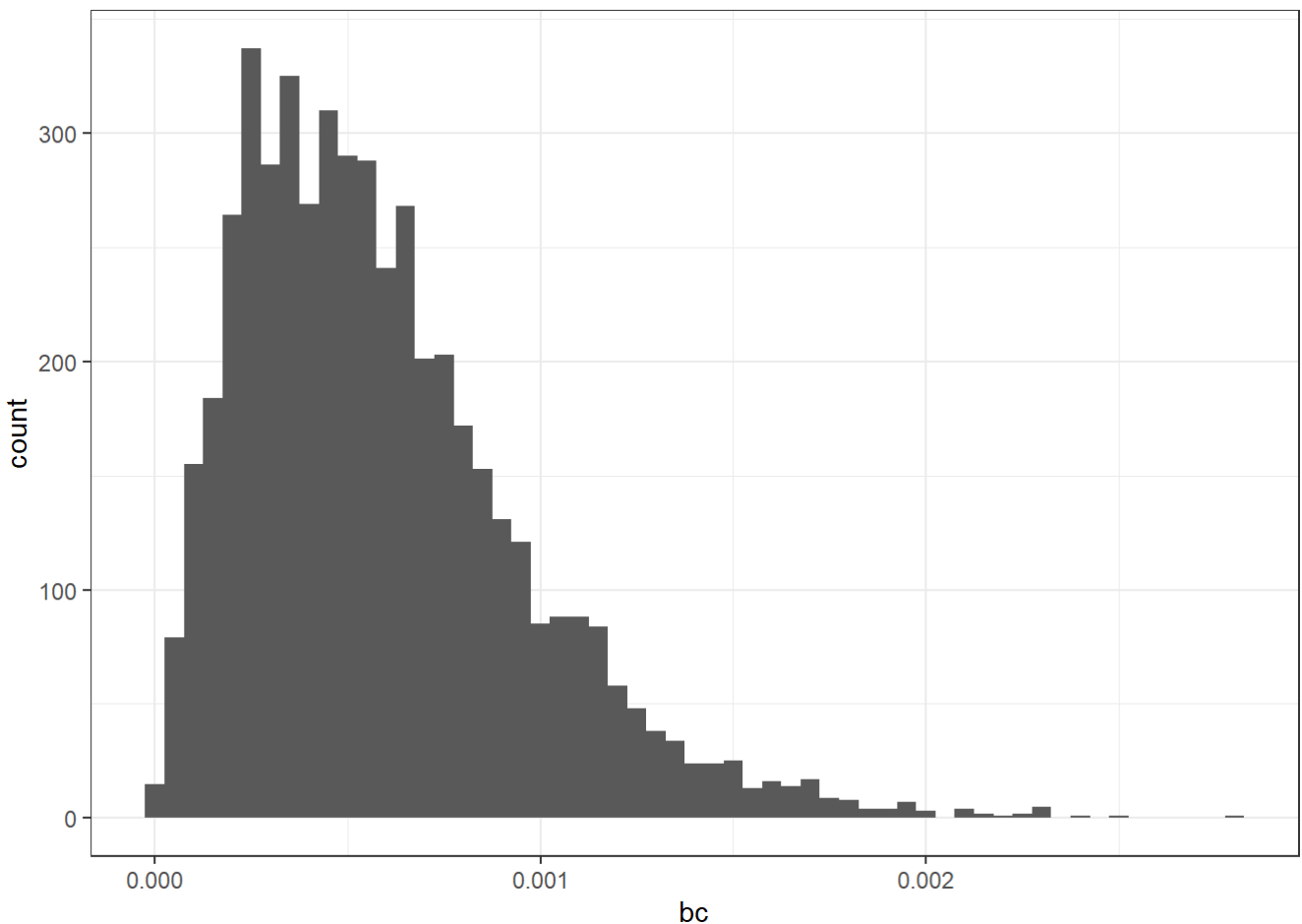
```
# calculate and compare NPV
npv.cc <- npv(sales.cc)
npvvec <- cbind(npvvec, npv.cc)
npvvec
```

```
##      npv.base npv.dc npv.ec npv.cc
## 1      3443    3446    3442    3417
```

中介中心性

首先，我们计算节点的中介中心性，并绘制中介中心性的分布图。

```
# calculate betweenness centrality
bc <- betweenness(g, normalized = T)
# distribution of bc
data.frame(bc = bc) %>% ggplot(aes(x = bc)) + geom_histogram(binwidth = 5e-5) + theme_bw()
```



进而，我们选取10个中介中心性最大的行动者（即消费者）作为“种子客户”，免费向其提供智能穿戴产品。

```
# select 2/1000 actors as seeds
seeds.bc <- order(bc, decreasing = T)[1:seedsizesize]
# print the bc of these seeds
bc[seeds.bc]
```

```
## [1] 0.00279 0.00250 0.00242 0.00231 0.00231 0.00230  
0.00229 0.00229  
## [9] 0.00227 0.00224
```

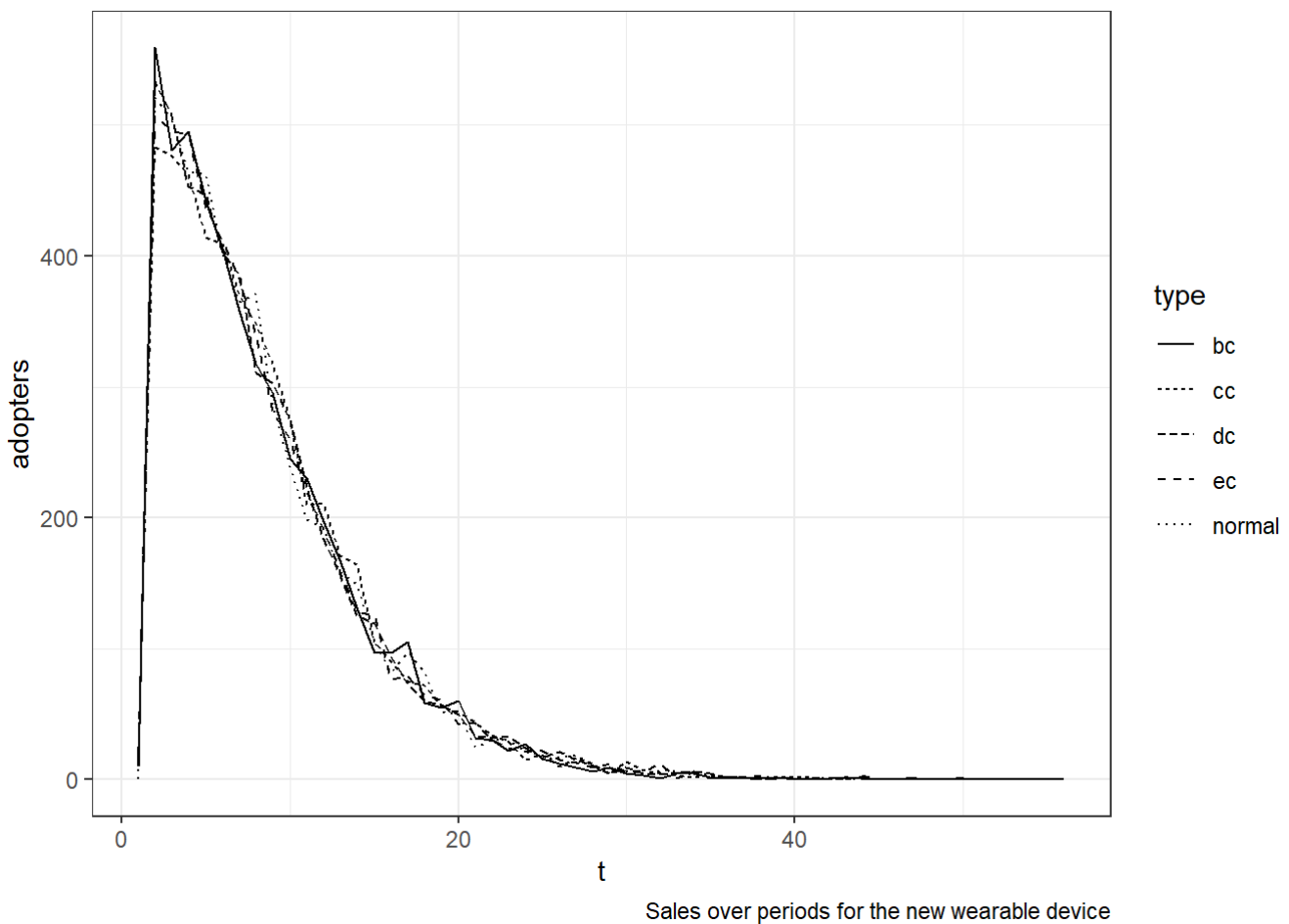
```
# set attributes
g.bc <- setattr(g, seeds = seeds.bc)
# diffusion process
g.bc <- iteration(g.bc)
# obtain the sales
sales.bc <- graph_attr(g.bc, "adopters")[[1]]

# create a data frame
t <- 1:max(length(sales.base$basesales), length(sales.d
c), length(sales.ec), length(sales.cc), length(sales.b
c))
# adding zeros
addzeros <- function(x, tmax) {
  if (length(x) < tmax) {
    # adding zeros
    x <- c(x, rep(0, tmax - length(x)))
  }
  # return
  return(x)
}
# apply for two vectors
illust.dat <- data.frame(t,
                          normal = addzeros(sales.base$b
asesales, length(t)),
                          dc = addzeros(sales.dc, length
(t)),
                          ec = addzeros(sales.ec, length
(t)),
                          cc = addzeros(sales.cc, length
(t)),
```

```

bc = addzeros(sales.bc, length
(t)))
illust.dat <- gather(illust.dat, "type", "adopters", 2:
6)
# plot the results
ggplot(illust.dat, aes(x = t, y = adopters, linetype =
type)) + geom_line() + labs(caption = "Sales over peri
ods for the new wearable device") + theme_bw()

```



可以看到，总销售量为5000。

最后，我们可以比较四种策略下的销售额净现值。

```

# calculate and compare NPV
npv.bc <- npv(sales.bc)
npvvec <- cbind(npvvec, npv.bc)
npvvec

```

##	npv.base	npv.dc	npv.ec	npv.cc	npv.bc
## 1	3443	3446	3442	3417	3460