

# Приближени пресмятания - подходи, методи, алгоритми

Статистическа обработка на данни с R

Пламен Петров и Тодор Балабанов

Център за обучение  
Институт по информационни и комуникационни технологии  
Българската академия на науките  
*p.petrov@iit.bas.bg todorb@iinf.bas.bg*

6.VI.2020

## Acknowledgments

These teaching materials are funded by Velbazhd Software LLC and it is partially supported by the Bulgarian Ministry of Education and Science (contract D01-205/23.11.2018) under the National Scientific Program “Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICTinSES)”, approved by DCM # 577/17.08.2018.

# Съдържание

## 1 Монте-Карло методи

- Концепция за Монте-Карло пресмятания
- Прилагане на Монте-Карло в R

## 2 Генетични алгоритми

- Обща информация
- Практически пример

## 3 Изкуствени неверонни мрежи

- Обща информация
- Практически пример

## 4 Заклучение

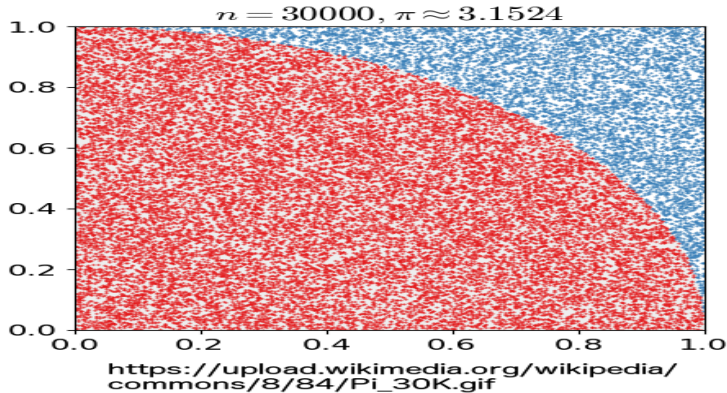
- Дискусия

# Монте-Карло методи

# Стъпки за изчисление

- Определяне на област от допустими стойности
- Генериране на извадка от случайни числа в предварително дефинираната област
- Извършване на точни пресмятания с така генерираните случайни числа
- Обобщаване на резултатите от извършените пресмятания

# Пресмятане на числото $\pi$ с Монте-Карло метод



## Стъпки по примера

- Изчертаване на квадрат и четвъртина окръжност вписана в него
- Генериране на случайни равномерно разпределени числа, като координати (х,у двойки) в описаната допустима област
- Изброяване на точките с координати х,у които са на разстояние 1 от центъра на четвърт окръжността
- Съотношението между точките в четвърт окръжността и общия брой точки е  $\pi/4$ , което умножено по 4 дава приближена стойност за числото  $\pi$

## Фактори силно влияещи на крайния резултат

- Ако генерираните случайни числа не са равномерно разпределени, това ще доведе до невярна апроксимация за търсената стойност
- Генерирането на малък брой координати за точки в дефиниционната област води до ниска надеждност на резултата от апроксимацията
- Колкото по-голям е броят на генерираните случайни числа, толкова по-надеждни са резултатите от апроксимацията



# Подготовка на извадката

## Генериране на данни от хвърляне на зарове

```
mode <- function(x) {
  values <- unique(x)
  return( values[which.max(tabulate(match(x, values)))] )
}
experiment <- function(n, d){
  x <- sample(6, n, TRUE)
  for(i in 2:d) {
    x <- x + sample(6, n, TRUE)
  }
  return( list(" mean" = mean(x), " median" = median(x),
    " mode" = mode(x)) )
}
```

# Извикване на функциите

## Изпълнение на симулацията

```
library( MonteCarlo )  
result <- MonteCarlo(func=experiment, nrep=1000,  
param_list=list("n"=c(10, 50, 100, 150, 200),"d"=c(1,2,6)))  
summary( result );  
MakeTable(output=result, rows=c("d"), cols=c("n","list"),  
digits=2, include_meta=FALSE)
```

# Сравнение на средна, медиана и мода за експеримент със зарове

list d/n	mean					median				
	10	50	100	150	200	10	50	100	150	200
1	10.50	10.48	10.49	10.50	10.50	10.52	10.50	10.48	10.50	10.52
2	7.01	7.00	7.00	7.01	6.99	7.02	7.01	7.00	7.00	7.00
6	20.94	21.03	21.00	21.01	21.00	20.90	21.01	20.99	21.00	21.00

# Генетични алгоритми

# Същност

- Инструмент за евристична глобална оптимизация
- Популационно организирани
- Процес на еволюция
- Основно приложими при задачи с голяма размерност на пространството от решенията

# Основни операции

- Селекция на родители
- Кръстосване
- Мутация
- Оценка

Възможно е да се приложи правило за елита.

# Задача за раницата - данни

Предмет	Ценност	Тегло
cake	10	1
ice cream	15	10
orange juice	10	5
strawberries	30	7
grape	30	1
candies	20	5
chocolate	2	1

# Структура от данни

## Организация на информацията

```
library( genalg )
data <- data.frame(item = c("cake", "ice cream", "orange juice",
"strawberries", "grape", "candies", "chocolate"), points = c(10,
15, 10, 30, 30, 20, 2), weight = c(1, 10, 5, 7, 1, 5, 1))
limit <- 20
solution <- c(1, 0, 0, 1, 1, 0, 0)
data[solution==1, ]
cat(solution %*% data$points)
```



# Оптималност на решенията

## Целева функция

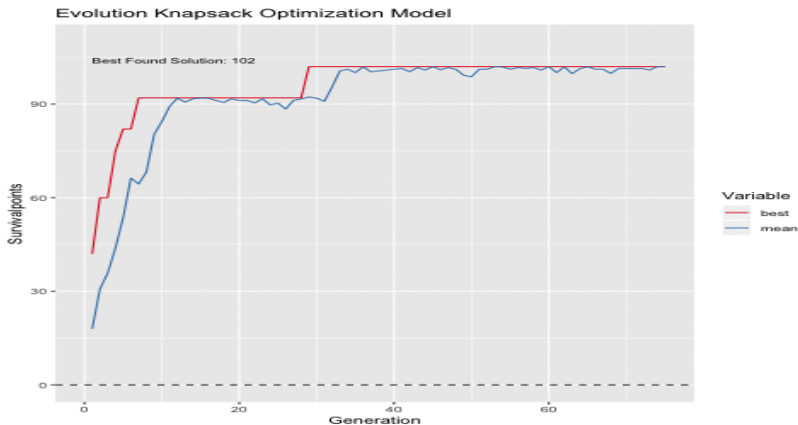
```
fitness <- function(x) {  
  points <- x %*% data$points  
  weight <- x %*% data$weight  
  if (weight > limit) {  
    return( 0 )  
  } else {  
    return( -points )  
  }  
}
```

# Извършване на пресмятанията

## Оптимизация на задачата за раницата с генетични алгоритми

```
iterations <- 75
model <- rbga.bin(size=7, popSize=37, iters=iterations,
mutationChance=0.01, elitism=TRUE, evalFunc=fitness)
cat( summary(model) )
best <- c(1, 0, 1, 1, 1, 1, 1)
data[best == 1, ]
cat(paste(best %*% data$points, " /", sum(data$points)))
```

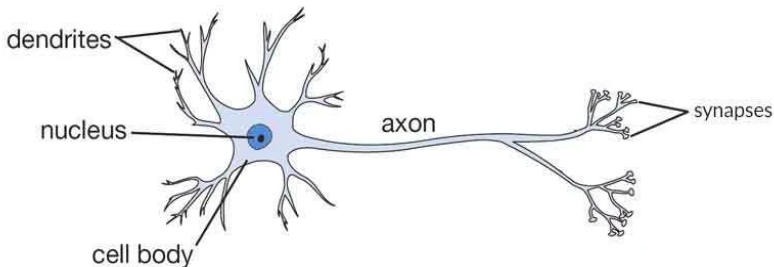
# Сходимость на процеса по оптимизация



# Изкуствени невронни мрежи

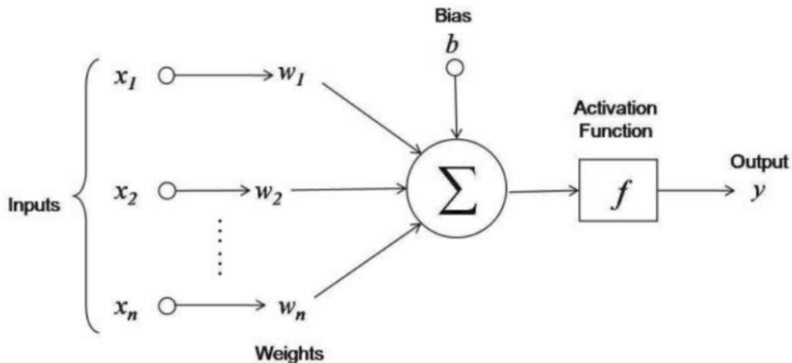
# Схема на биологична нервна клетка

## Biological Neuron



[https://res.cloudinary.com/dyd911kmh/image/upload/f\\_auto,q\\_auto:best/v1547672259/1\\_a74o1a.png](https://res.cloudinary.com/dyd911kmh/image/upload/f_auto,q_auto:best/v1547672259/1_a74o1a.png)

# Схема на изкуствен неврон



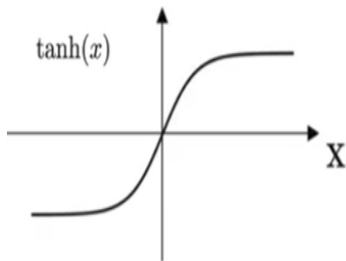
[https://res.cloudinary.com/dyd911kmh/image/upload/f\\_auto,q\\_auto:best/v1547672259/2\\_i1cdwq.png](https://res.cloudinary.com/dyd911kmh/image/upload/f_auto,q_auto:best/v1547672259/2_i1cdwq.png)

# Трансферна функция в изкуствен неврон

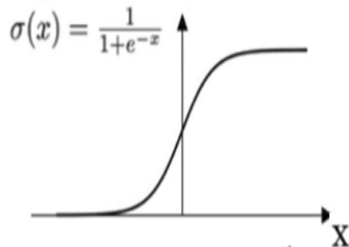
$$y_i = \sum (w_{ij} * y_j) + b \quad (1)$$

# Функции за активация на изкуствен неврон

## Hyper Tangent Function



## Sigmoid Function







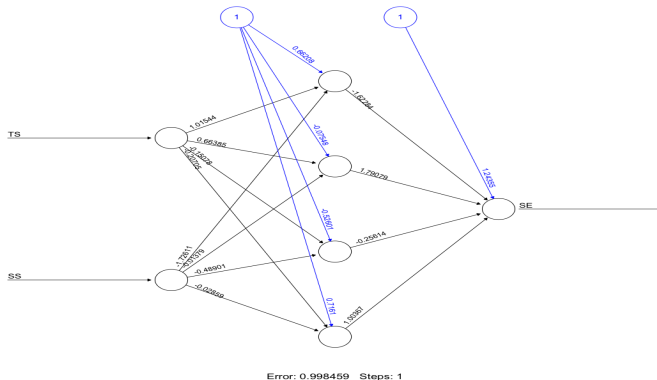
# Обучение с учител

## Класифициране с изкуствена невронна мрежа

```
library( neuralnet )
training <- data.frame(TS=c(30,10,20,30,80),
SS=c(80,20,90,40,50), SE=c(1,0,1,0,1))
network <- neuralnet(SE TS+SS, data=training, hidden=4,
act.fct="tanh", linear.output=FALSE)
plot( network )
testing <- data.frame(TS=c(85,30,20), SS=c(10,20,75))
prediction <- compute(network, testing)
prediction$net.result
ifelse(prediction$net.result>0.5, TRUE, FALSE)
```

## Практически пример

## Трислойна изкуствена невронна мрежа



## Тестово множество данни

Technical Skills	Soft Skills	Employment Forecast
85	10	0.3322677
30	20	0.3322677
20	75	0.9924267

# Заклучение

## Въпроси и отговори

Благодаря за вниманието!