北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

Data Analytics with R
语言数据分析

既是世间法、自当有分别

艾新波 / 2018 • 北京

课程体系

- R语言数据分析
  - 上部：论道 
    - 第1章 气象万千、数以等观
    - 第2章 所谓学习、归类而已
    - 第3章 格言联璧话学习
    - 第4章 源于数学、归于工程
  - 中部：执具 
    - 第5章 工欲善其事必先利其器
    - 第6章 基础编程
    - 第7章 数据对象
  -
    - 第8章 人人都爱tidyverse
    - 第9章 最美不过数据框
  - 下部 博术 
    - 第10章 观数以形
    - 第11章 相随相伴、谓之关联
    - 第12章 既是世间法、自当有分别
    - 第13章 方以类聚、物以群分
    - 第14章 庐山烟雨浙江潮

R语言实现

```
library(kknn)
set.seed(2012)
imodel <- kknn(wlfk ~ .,
                    train = cjb[train_set_idx, ],
                    test = cjb[train_set_idx, ])
predicted_train <- imodel$fit
#ce: classification error
Metrics::ce(cjb$wlfk[train_set_idx], predicted_train)
#> [1] 0.1090573
```

R语言实现

```
#作为惰性学习法，训练和测试同时进行
imodel <- kknn(wlfk ~ .,
                    train = cjb[train_set_idx, ],
                    test = cjb[-train_set_idx, ])
predicted_test <- imodel$fit
Metrics::ce(cjb$wlfk[-train_set_idx], predicted_test)
#> [1] 0.1888412
```

R语言实现

```
#选取最优的k和核
train_kk <- train.kknn(
  wlfk ~ .,
  data = cjb,
  kmax = 100,
  kernel = c(
    "rectangular", "epanechnikov",
    "cos", "inv",
    "gaussian", "optimal"))
```

R语言实现

#查看具体结果

```
train_kk
#> Call: train.kknn(formula = wlfk ~ ., data = cjb,
#>     kmax = 100, kernel = c("rectangular",
#>       "epanechnikov", "cos", "inv", "gaussian", "optimal"))
#>
#> Type of response variable: nominal
#> Minimal misclassification: 0.2105943
#> Best kernel: gaussian
#> Best k: 49
```

R语言实现

```
#最佳的k值
best_k <- train_kk$best.parameters$k
best_k
#> [1] 49

best_kernel <- train_kk$best.parameters$kernel
best_kernel
#> [1] "gaussian"
```

R语言实现

```
#提取不同k和核相应的分类错误率
ce_kk <- train_kk$MISCLASS
View(ce_kk)
#最小错误率
min_ce <- min(ce_kk)
```

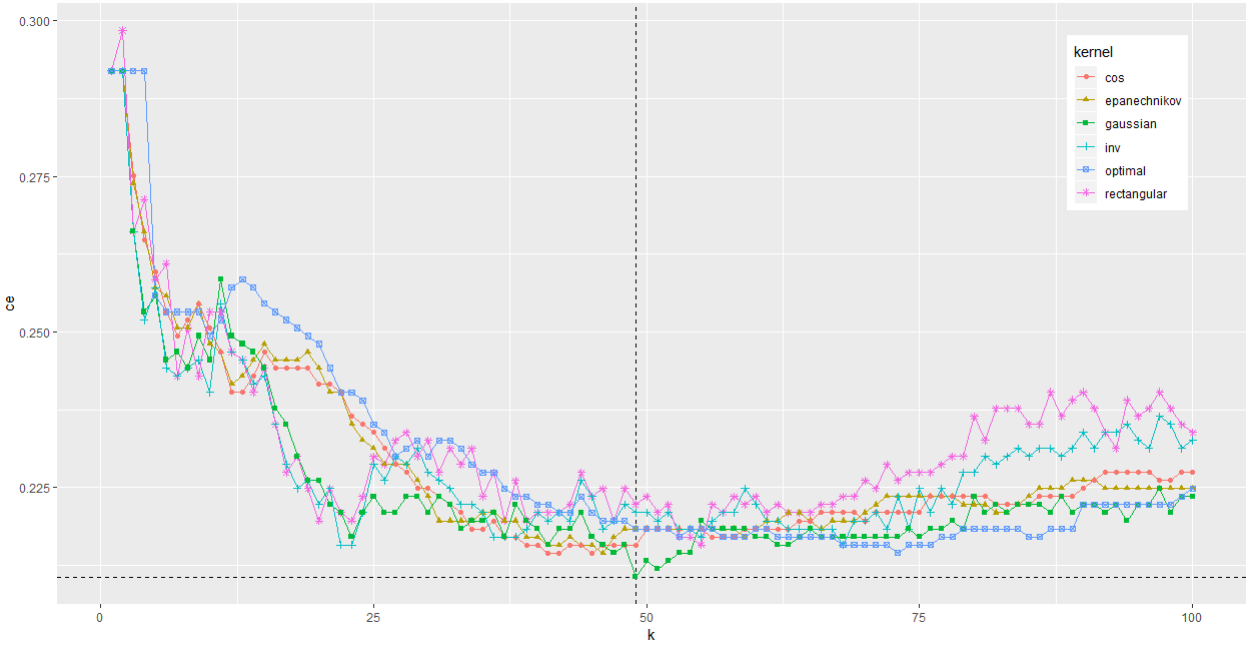
	rectangular	epanechnikov	cos	inv	gaussian	optimal
1	0.291	0.291	0.291	0.291	0.291	0.291
2	0.298	0.291	0.291	0.291	0.291	0.291
3	0.257	0.273	0.274	0.257	0.257	0.291
4	0.269	0.266	0.266	0.244	0.247	0.291
5	0.258	0.257	0.258	0.257	0.256	0.252
6	0.261	0.257	0.256	0.242	0.242	0.244
7	0.240	0.242	0.245	0.239	0.245	0.243
8	0.251	0.245	0.244	0.242	0.240	0.243
9	0.251	0.248	0.247	0.253	0.252	0.249
10	0.247	0.245	0.243	0.242	0.245	0.253
11	0.245	0.243	0.242	0.247	0.249	0.253
12	0.240	0.238	0.235	0.239	0.240	0.253
13	0.239	0.243	0.238	0.238	0.242	0.248
14	0.243	0.243	0.240	0.242	0.240	0.248
15	0.242	0.244	0.244	0.239	0.243	0.248
16	0.236	0.244	0.244	0.234	0.235	0.249
17	0.235	0.245	0.243	0.234	0.233	0.249
18	0.231	0.245	0.244	0.230	0.234	0.251
19	0.233	0.244	0.244	0.231	0.234	0.244
20	0.220	0.242	0.242	0.226	0.229	0.244

Showing 1 to 20 of 100 entries

R语言实现

```
#通过ggplot2进行绘制
as.data.frame(ce_kk) %>%
  mutate(k = 1:nrow(ce_kk)) %>%
  gather(key = "kernel", value = "ce", -k) %>%
  ggplot(aes(x = k, y = ce, colour = kernel)) +
  geom_vline(aes(xintercept = best_k), linetype = "dashed") +
  geom_hline(aes(yintercept = min_ce), linetype = "dashed") +
  geom_line() +
  geom_point(aes(shape = kernel)) +
  theme(legend.position = c(0.9,0.8))
```

R语言实现


0.300
0.275
0.250
0.225
ce
0
25
50
75
100
k
kernel
cos
epanechnikov
gaussian
inv
optimal
rectangular

R语言实现

```
library(kknn)
sp <- Sys.time() #记录开始时间
cat("\n[Start at:", as.character(sp))
for (i in 1:length(kfolds)) {
  curr_fold <- kfolds[[i]] #当前这一折
  train_set <- cjb[-curr_fold,] #训练集
  test_set <- cjb[curr_fold,] #测试集
  predicted_train <- kknn(
    wlfk ~ ., train = train_set, test = train_set,
    k = best_k, kernel = best_kernel)$fit
```

R语言实现

```
  imetrics("kknn", "Train", predicted_train, train_set$wlfk)
  predicted_test <- kknn(
    wlfk ~ ., train = train_set, test = test_set,
    k = best_k, kernel = best_kernel)$fit
  imetrics("kknn", "Test", predicted_test, test_set$wlfk)
}
ep <- Sys.time()
cat("\tFinised at:", as.character(ep), "]\n")
cat("[Time Ellapsed:\t",
   difftime(ep, sp, units = "secs"), " seconds]\n")
```

R语言实现

```
#>      method  type  accuracy error_rate
#> 1     knn Train 0.8333333  0.1666667
#> 2     knn  Test 0.8076923  0.1923077
#> 3     knn Train 0.8405172  0.1594828
#> 4     knn  Test 0.8076923  0.1923077
#> 5     knn Train 0.8333333  0.1666667
#> 6     knn  Test 0.8461538  0.1538462
.....
#> 19    knn Train 0.8278336  0.1721664
#> 20    knn  Test 0.7792208  0.2207792
```

究竟是一种什么关系

既然所有规律都是关系

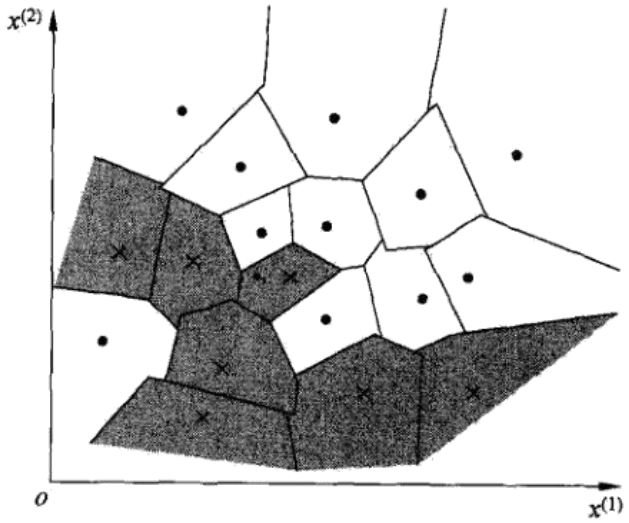
那么，请问：

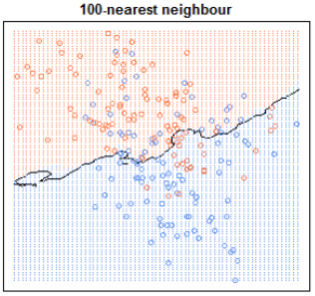
近邻法

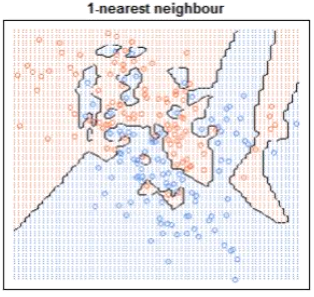
究竟是什么关系

得到规律的表现形式是什么

近邻法：空间划分的角度







左图引自: 李航. 统计学习方法. 北京：清华大学出版社, pp.38

谢谢聆听

Thank you

教师个人联系方式

艾新波

手机：13641159546

QQ： 23127789

微信：13641159546

E-mail: 13641159546@126.com

axb@bupt.edu.cn

地址：北京邮电大学科研楼917室

课程网址：https://github.com/byaxb/RDataAnalytics