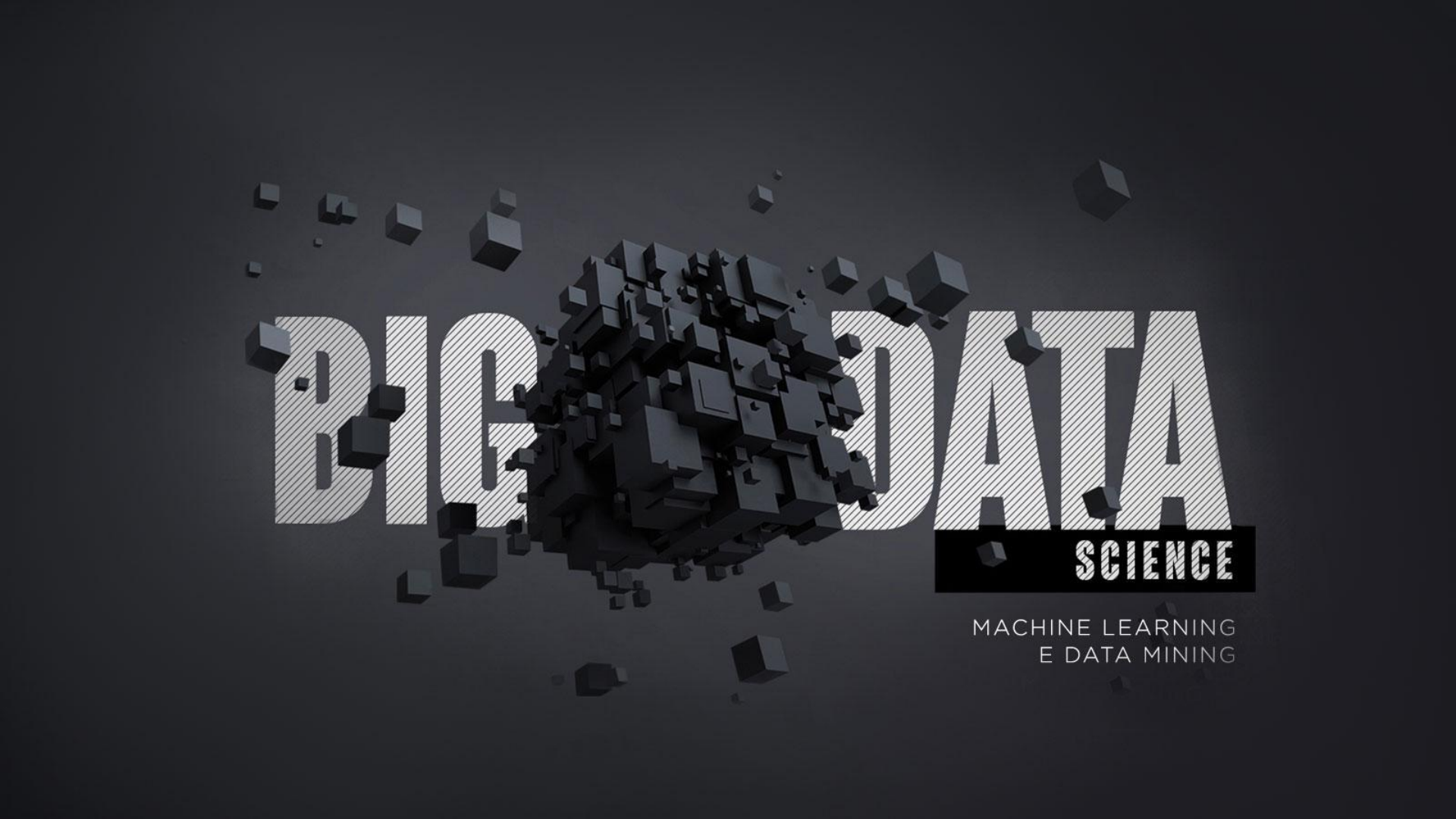

BIG DATA
SCIENCE
MACHINE LEARNING
E DATA MINING

BIG DATA SCIENCE

PARTE 4

DIÓGENES JUSTO

D3.JS

É uma biblioteca em javascript utilizada para manipular documentos baseada em dados.

Por ser um padrão aberto, pode ser acoplada facilmente a outras ferramentas e browsers de última geração.

EXEMPLOS:

http://projects.delimited.io/experiments/d3-threejs/

http://humancost.info/

ScatterPlot

D3

```
install.packages("scatterD3")

library(scatterD3)
```

Utiliza a D3.js para criar gráficos de dispersão mais ricos e dinâmicos.

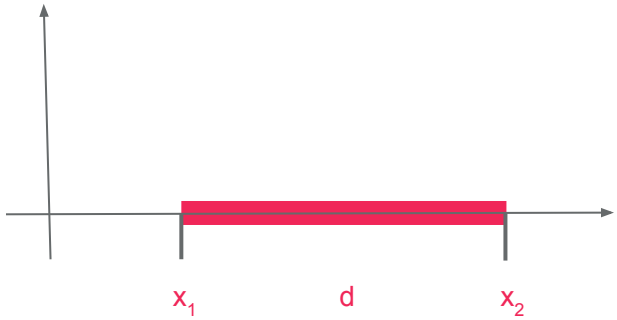
```
d <- mtcars

scatterD3(x = d$wt, y = d$mpg, lab = rownames(d),

col_var=d$cyl, symbol_var = d$am, symbol_lab =

"Manual transmission")
```

QUAL É O SEU CONCEITO DE SIMILARIDADE?

DISTÂNCIA ENTRE DOIS PONTOS

Para o caso unidimensional:

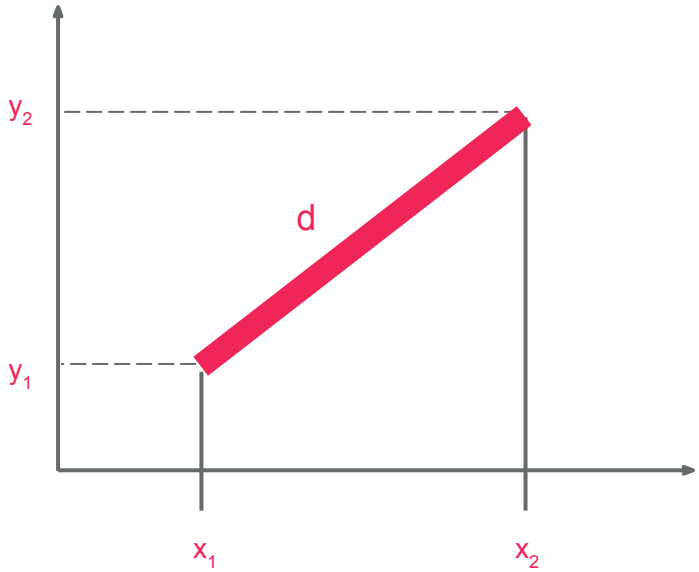
$$d = x_2 - x_1$$



DISTÂNCIA ENTRE DOIS PONTOS

E para o caso bidimensional?

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$



CLUSTERIZAÇÃO

- UMA COLEÇÃO DE OBJETOS COM:
  - Alta similaridade (relacionados) no grupo
  - Baixa similaridade (não relacionados)
- APRENDIZAGEM NÃO SUPERVISIONADA (sem classes pré-definidas):
  - Contrasta com classificação (supervisionada)
- APLICAÇÕES COMUNS:
  - Análise de distribuições
  - Etapa intermediária (precede aprendizagem supervisionada, por exemplo)
  - Identificação de outliers

SIMILARIDADE

- CONCEITO DE SIMILARIDADE:
  - Dois objetos são similares se eles te um grau alto de proximidade, isto é, pequena distância na classe que está sendo analisada

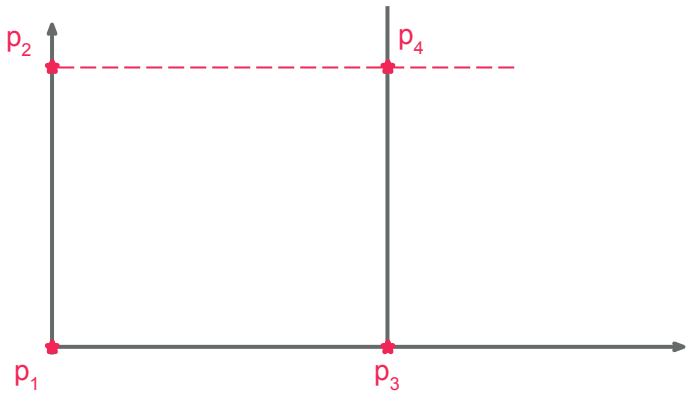
- EXEMPLO:
  - Alunos estudiosos ou não: distância da nota
  - Pessoas parecidas: distância entre pontos da face
  - Detecção de movimento: distância entre similaridade de quadros (frames) aumentou

SIMILARIDADE

- FUNÇÃO SIMILARIDADE (EM GERAL):
  - Número real
  - Entre 0 (completamente diferentes) e 1 (idênticos)

- DISSIMILARIDADE (DISTÂNCIA)
  - Entre 0 (idênticos) e 1 (completamente diferentes)

DISTÂNCIA NUMÉRICA



Ponto	X	Y
p1	0	0
p2	0	1
p3	1	0
p4	1	1

Matriz de dados

	p1	p2	p3	p4
p1	0			
p2	1	0		
p3	1	1,41	0	
p4	1,41	1	1	0

Matriz de dissimilaridade
(distância Euclidiana)

DISTÂNCIA NUMÉRICA: MINKOWSKI

DISTÂNCIA (OU MÉTRICA) DE MINKOWSKI*:

$$d(i,j) = \sqrt[p]{|x_{i1} - x_{j1}|^p + |x_{i2} - x_{j2}|^p + \ldots + |x_{il} - x_{jl}|^p}$$

PROPRIEDADES:

d(i,j) > 0, se i<>j e d(i,i)=0	(Positividade)
d(i,j) = d(j,i)	(Simetria)
d(i,j) <= d(i,k) + d(k,j)	(Desig. do triângulo)

Casos específicos de p (chama-se norma L-p)

Norma L-1: distância de Manhattan
Norma L-2: distância Euclidiana
Norma L-max ou L-inf: distância "Suprema"

OUTRAS DISTÂNCIAS

- DISTÂNCIA PARA ATRIBUTOS BINÁRIOS:
  - Coeficiente de Jaccard (coerência)
- Distância para ATRIBUTOS CATEGÓRICOS
- Distância entre VARIÁVEIS ORDINAIS
- SIMILARIDADE do cosseno (utilizado em text mining)

MÉTODO K-MEANS*

Cada cluster é representado por um centro (centróide)
Arbitra-se K clusters, inicie:

- Aleatoriamente escolha K centróides
- Faça até critério de convergência
    - Associe cada ponto ao centróide mais próximo
    - Para cada cluster, calcule a nova posição do centróide como ponto médio do cluster

VARIAÇÕES DO K-MEANS*

Melhor escolha dos K centróides iniciais

- K-Means++, K-Means inteligente

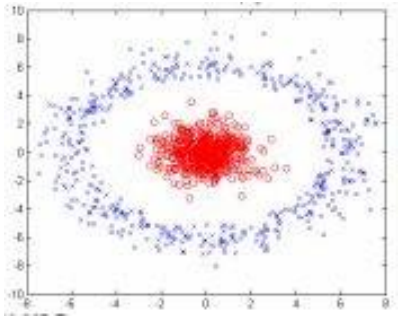
Diferentes tipos de clusters

- K-Medois, K-Medians, K-Modes

Aplicando tranformações

- K-Means ponderado, Kernel K-Means

OUTRAS DISTÂNCIAS



- K-Means++: convergência mais rápida, resolve em parte problema de qualidade de clusters.
- K-Medoids/K-Median: menos sensível a outliers.
- Kernel K-Means: clusters em regiões não lineares

K-MEANS

Veja uma animação que mostra o funcionamento do K-Means:

http://shabal.in/visuals/kmeans/1.html

EXERCÍCIO 2

Clusters para MPG x WT

```
m <- mtcars
k <- kmeans(m[,c("mpg", "wt")], 3 )
m$k <- k$cluster
plot(m$mpg~m$wt, col=m$k)
```

Obs:

k$cluster -> número de cada cluster para cada ponto

k$centers -> centróide de cada cluster

A clusterização como etapa prévia de processamento, gerando uma nova característica (variável) categórica.

```
summary(lm(mpg~wt), data=m)
summary(lm(mpg~wt + k, data=m)
```

REVISÃO DO CONTEÚDO

EXERCÍCIO REVISÃO

ÁRVORES DE DECISÃO COM TITANIC

```
library(party)
df <- read.csv("train.csv")
set.seed(33)
df<- df[sample(nrow(df)) ,]
train <- df[1:600,]
test <- df[601:891,]
mDT <- ctree(Survived ~ Sex,data=train)
pred <- predict(mDT, newdata=test)
table(pred, test$Survived)
```

ÁRVORES DE DECISÃO COM TITANIC

```
mDT <- ctree(Survived ~ Sex+Age+Pclass+Fare, data=
train)
pred <- predict(mDT, newdata=test)
table(pred, test$Survived)
```

ÁRVORES DE DECISÃO COM TITANIC

```
mDT <- ctree(Survived ~ Sex+Age+Pclass+Fare, data=
train)
pred <- predict(mDT, newdata=test)
table(pred, test$Survived)
```

```
#Regressor x Classificador: as.factor
endógena
df$Survived_F <- as.factor(df$Survived)
```

VARIÁVEL CATEGÓRICA

NO R UTILIZAMOS A CLASSE FACTOR

Função factor() factor(c(1,2,3,3,3,1))

Como inserir dados em uma variável factor?

```
v1<-1:6
  v2<-factor(c(1,2,3,3,3,1))
    d<-data.frame(cbind(v1, v2))
 str(d)
    d$v2 <- as.factor(d$v2)
   rbind(d,c(7,4))
```

NO R UTILIZAMOS A CLASSE FACTOR

Inserindo uma nova categoria (level)

```
levels(d$v2)

levels(d$v2) <- c(levels(d$v2), "4")

levels(d$v2)

rbind(d,c(7,4))
```

AVALIANDO E COMPARANDO MODELOS DE CLASSIFICAÇÃO

AVALIAÇÃO DE MODELOS

Como avaliar se um modelo prevê corretamente os dados?

REAL		PREVISTO	
		PVC	NORMAL
	PVC	0.78	0.22
	NORMAL	0.16	0.84

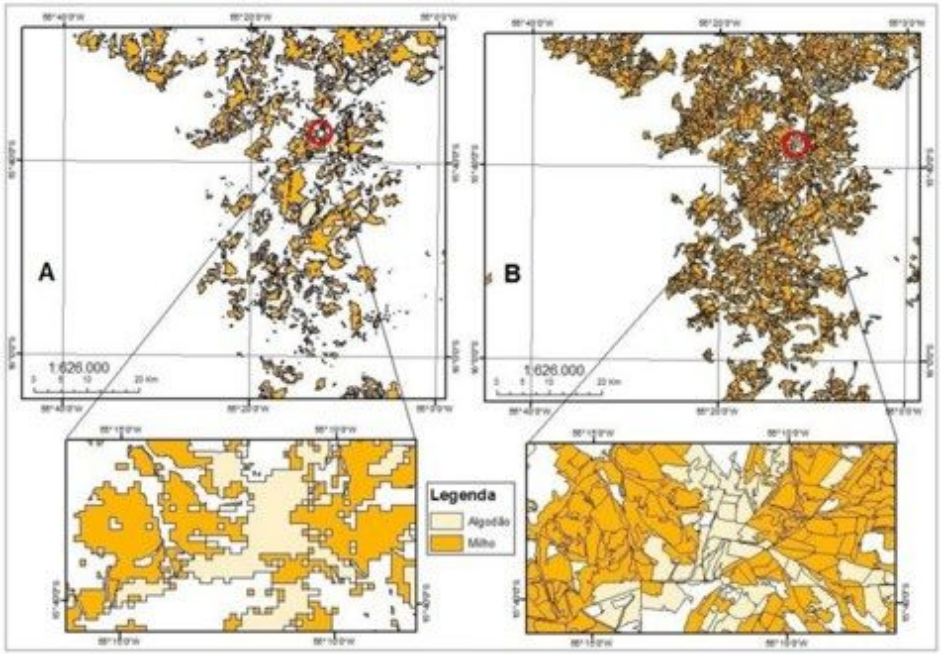
MATRIZ DE CONFUSÃO DOS RESULTADOS

AVALIAÇÃO DE MODELOS

Obs.1: na fonte não é citado se o real é coluna ou linha.
Inseri arbitrariamente para utilizar o exemplo.

Obs.2: reparem que os dados foram apresentados como percentual (probabilidade) em linha. Isto é, 78% dos PVCs reais foram previsto corretamente. 22% foram estimados erroneamente como normais.

Fonte: http://slideplayer.com.br/slide/1266306/
Classificação de arritmias cardíacas



Fonte: MAPEAMENTO DE CULTURAS AGRÍCOLAS POR IMAGENS DE SATÉLITE - COSTA, SOUTO e Zeilhofer

http://www.sinageo.org.br/2012/trabalhos/10/10-379-577.html

		VERDADE DE CAMPO			
CLASSIFICADOR		ALGODÃO	MILHO	OUTROS	TOTAL
	ALGODÃO	5.569	819	0	6.388
	MILHO	0	3.552	0	3.552
	OUTROS	0	761	5.674	6.435
	TOTAL	5.569	5.119	5.674	16.375

Arv Decisão

```
~/Dropbox/teaching/titanic/
> # Matriz de confusão
> table(prev, teste$Sur

prev   0   1
   0 162  64
   1   5  60
>
```

Regr Logist

```
          Reference
Prediction   0   1
         0 110  43
         1  13  58
```

MODELOS DE CLASSIFICAÇÃO – AVALIAÇÃO

Confusion Matrix para Titanic:

- 891 passageiros (em Train)
- Acerto 78% => ? Passsageiros?
- ~ 326

x1 + x2 + x3 = ?

		PREVISTO (Estimado no modelo)	
		PVC	NORMAL
REAL	SURVIVED	326	x1
	NOT SURVIVED	x2	x3

MODELOS DE CLASSIFICAÇÃO – AVALIAÇÃO

Como avaliar se um modelo prevê corretamente os dados?

- Matriz de erro ou Confusion Table

	CONDIÇÃO POSITIVA	CONDIÇÃO NEGATIVA
TESTE POSITIVO	TP (True positive)	FP (False Positive - Erro 2)
TESTE NEGATIVO	FN (False Negative– Erro1)	TN (True Negative)

MODELOS DE CLASSIFICAÇÃO – MÉTRICAS

Métricas comparativas:

	CONDIÇÃO POSITIVA	CONDIÇÃO NEGATIVA
TESTE POSITIVO	TP (True positive)	FP (False Positive)
TESTE NEGATIVO	FN (False Negative)	TN (True Negative)

ACC (Accuracy) : ACC = (TP + TN)/Tot

- % acertos do total

Precision: P = TP / (TP + FP)

- % acertos das minhas "afirmativas"

Sensitivity (Recall): R = TP / (TP + FN)

- % acertos frente a condição positiva (testa presença da característica em elementos que se assume que tenham)

MODELOS DE CLASSIFICAÇÃO – MÉTRICAS

Métricas comparativas:

	CONDIÇÃO POSITIVA	CONDIÇÃO NEGATIVA
TESTE POSITIVO	TP (True positive)	FP (False Positive)
TESTE NEGATIVO	FN (False Negative)	TN (True Negative)

Specificity = TN / (FP + TN)

- % de True Negative

(testa ausência da característica em elementos que se assume que não a tenham)

F1 = 2*TP / (2*TP + FP + FN)

- Métrica geral para comparar modelos

CURVA ROC

http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-89102006000600021

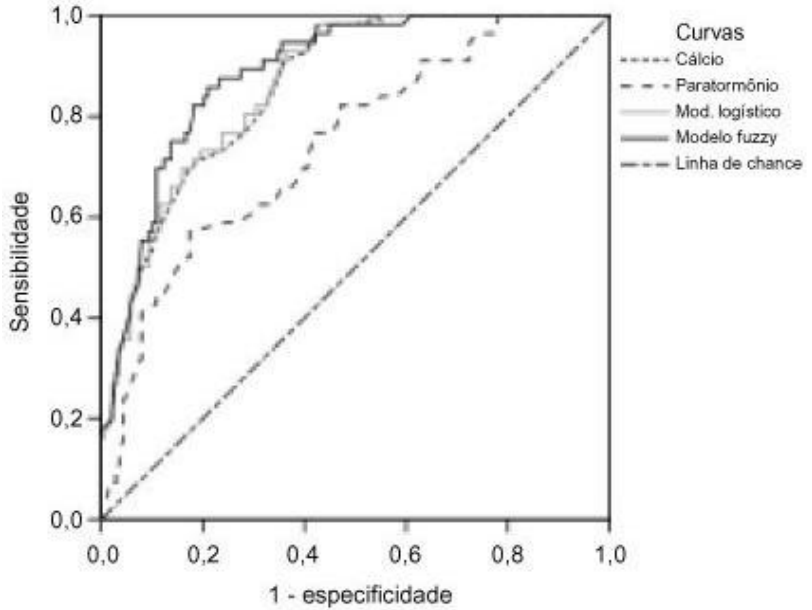


Figura 3 - Curvas ROC: Desempenho de cálcio, paratormônio, modelo logístico e modelo fuzzy na indicação de cintilografia das paratiróides.

MAIS ALGORITIMOS REGRESSÃO LOGÍSTICA

REGRESSÃO LOGÍSTICA

- É um Modelo de Regressão Generalizado (ou MLG, modelo linear generalizado), assim como regressão de Poisson, entre outros
- Utiliza, assim como demais MLG, um algoritmo que aplica o método da máxima verossimilhança (proposto por Nelder e Wedderburn)

REGRESSÃO LOGÍSTICA

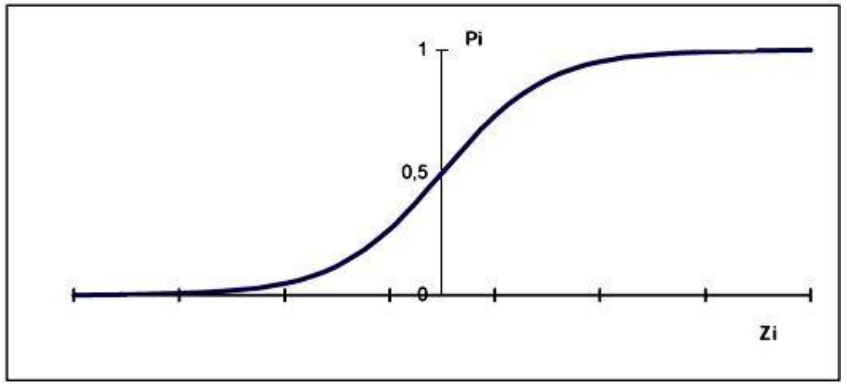


Figura 1 – Função logística acumulada

REGRESSÃO LOGÍSTICA

- Trabalha bem como classificador
- Tem o seguinte formato: $y = \frac{1}{1 + e^{-f(x)}}$
- Como modelos uma função linear (f(X)=ax+b) basta substituir na fórmula acima e temos sua formulação clássica.

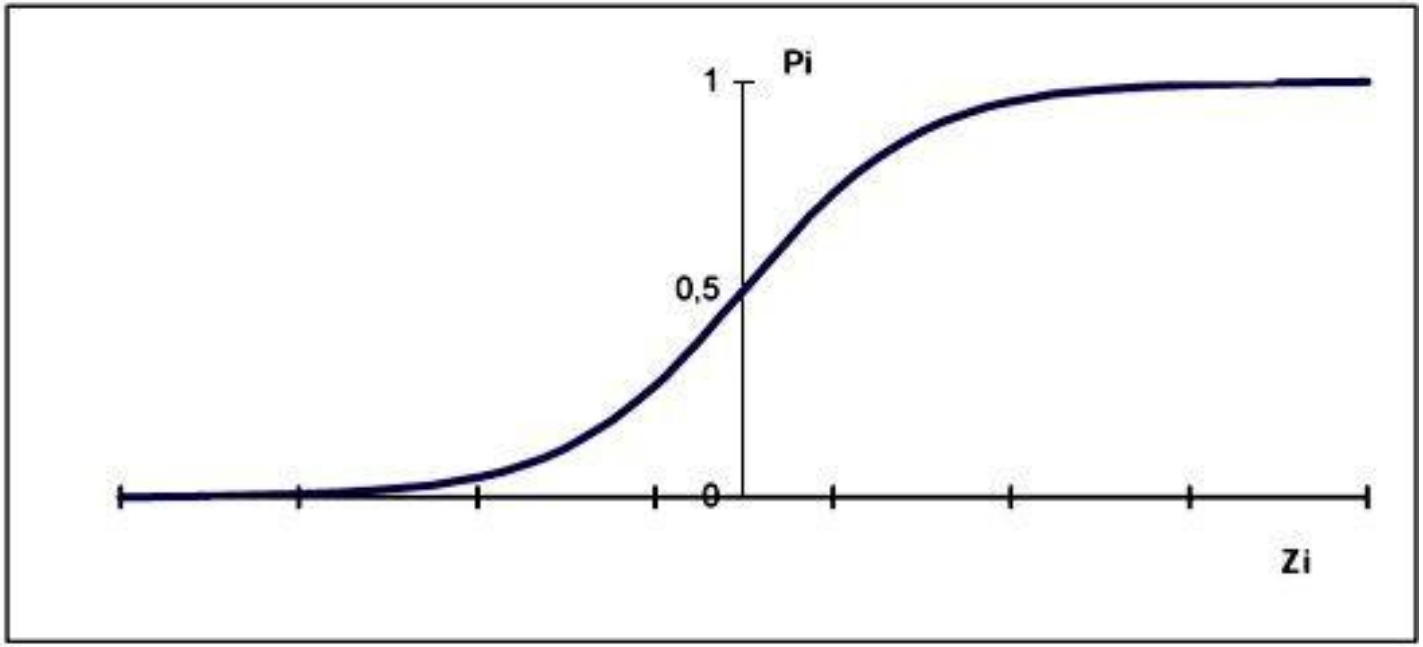


Figura 1 – Função logística acumulada

EXERCÍCIO

TITANIC COM REGRESSÃO LOGÍSTICA

```
install.packages("e1071")
library(e1071)
mRL <- glm(Survived_F~Sex+Age+Pclass+Fare, data=
train, family = binomial())

pred <- predict(mRL, newdata=test)
table(pred, test$Survived)

table(ifelse(pred<0,0,1), test$Survived)
```

MAIS ALGORITMOS
RANDOM FOREST

Uma evolução da técnica de árvores de decisão, que procura resolver o problema de overfit. Criado por Brieman & Cutler é uma técnica mista (ensemble learning) que compreende entre outras técnicas prune (poda) e bagging (bootstrap aggregating).

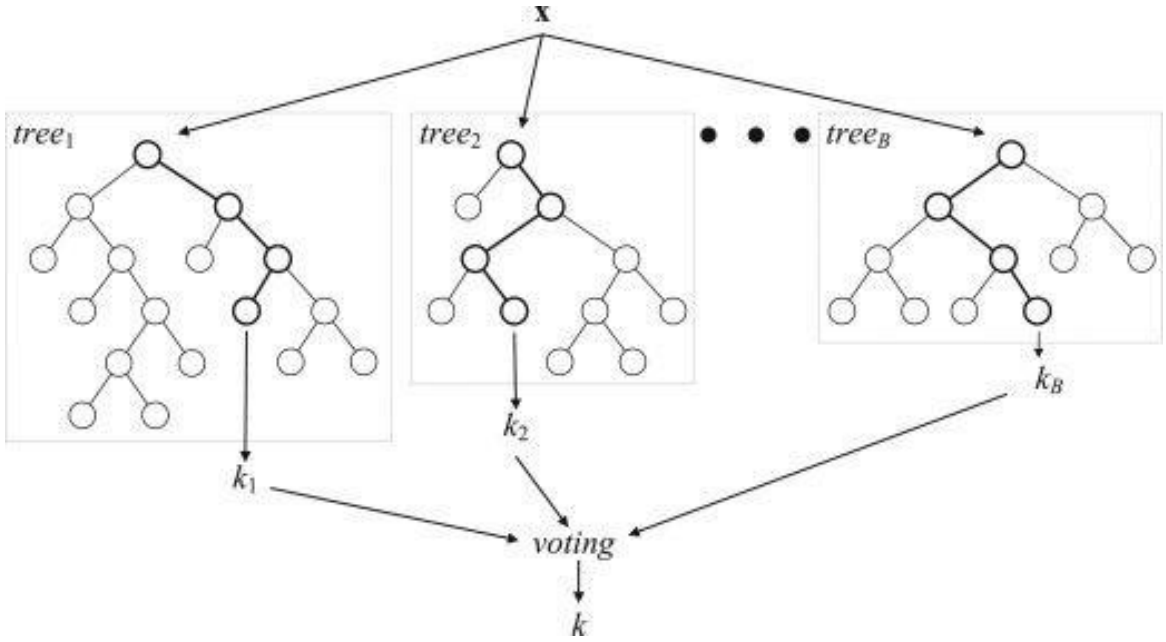
Em geral:

- Corrige overfit de árvores de decisão
- Trabalha bem com pequenas amostras
- Tem boa precisão com test-sets e real-sets diferentes do train-set

Referências: Brieman & Cutler (Random Forest(™) é uma marca registrada deles) http://jmlr.csail.mit.edu/

CONCEITO DE FUNCIONAMENTO

elaboração de um grande número de árvores de decisão (uma floresta), escolhidas aleatoriamente. É estabelecido de um critério de decisão para escolha do melhor resultado.


x
tree1
tree2
treeB
k1
k2
kB
voting
k

TÉCNICAS UTILIZADAS

- Validação do erro com 1/3 da amostra (out of bag);
- Avaliação da relevância das variáveis para escolha;
- Avaliação da importância em relação a hierarquia;
- Execução de interações aleatórias;
- Gera os "votos" nos ramos pela avaliação de proximidade (similar ao K-Nearest, K-NN);

TÉCNICAS UTILIZADAS

- Usa o cálculo de auto-funções (matricial) para acelerar em escala (grande volume de dados);
- Analisa outliers;
- Faz balanceamento de erros e trata problemas com fitting

EXERCÍCIO

TITANIC COM RANDOM FOREST

```
install.packages("randomForest")
library(randomForest)
mRF <- randomForest(Survived_F~Sex+Age+Pclass+Fare,
data=train, importance=TRUE, proximity=TRUE)

pred <- predict(mRF, newdata=test)
table(pred, test$Survived)
```

MAIS ALGORITIMOS
SVM

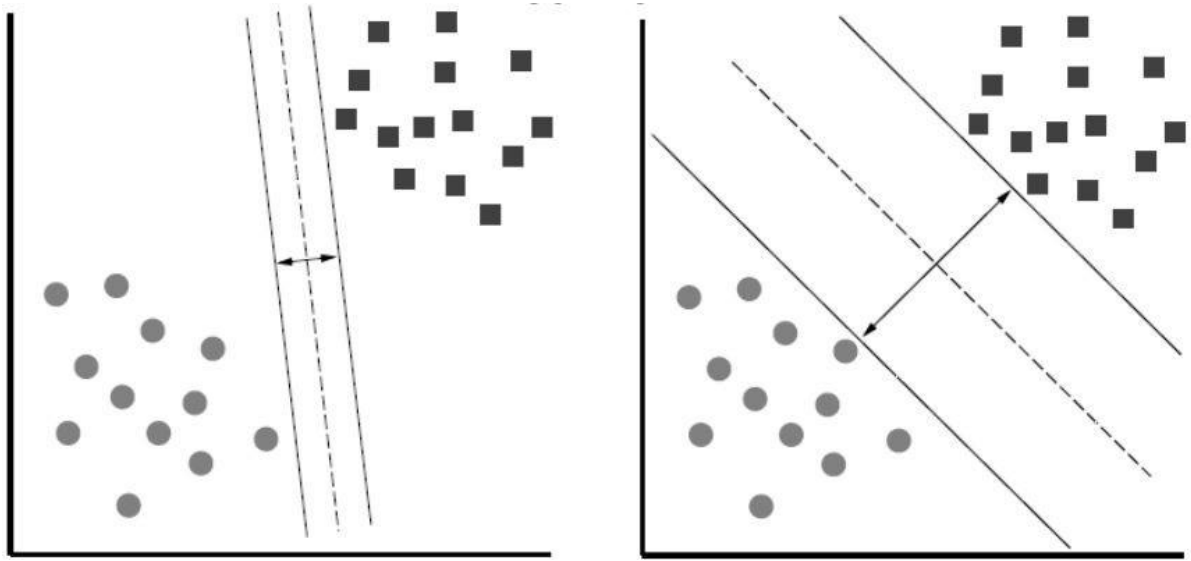
SVM
SUPPORT VECTOR MACHINES

Como separar dois conjuntos de dados por uma linha? E da forma ótima?

Conceito de margem: max (distância entre conjuntos)

http://www.svms.org/tutorials/Berwick2003.pdf

SVM - SUPPORT VECTOR MACHINES



É uma "nova geração" de algoritmos, que apresenta evoluções em relação ao uso de redes neurais e outros (classificadores):

- Utilizado para classificação de SPAM's de emails (junto com o classificador de Naive Bayes)

- "Estado da arte" para utilização em Text Mining

PONTOS FORTES

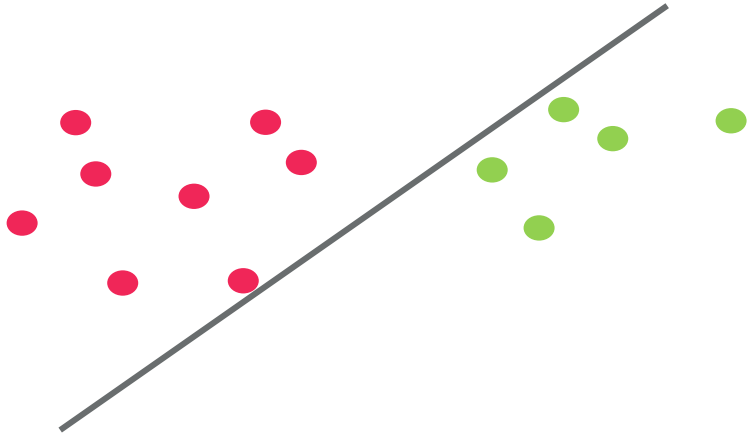
- Não necessita percorrer todos os pontos, somente a "fronteira" da linha (ou hiperplano, plano n-dimensional) de decisão: tem melhor performance sobre grandes volumes.

ENCONTRAR A, B E C TAIS QUE:

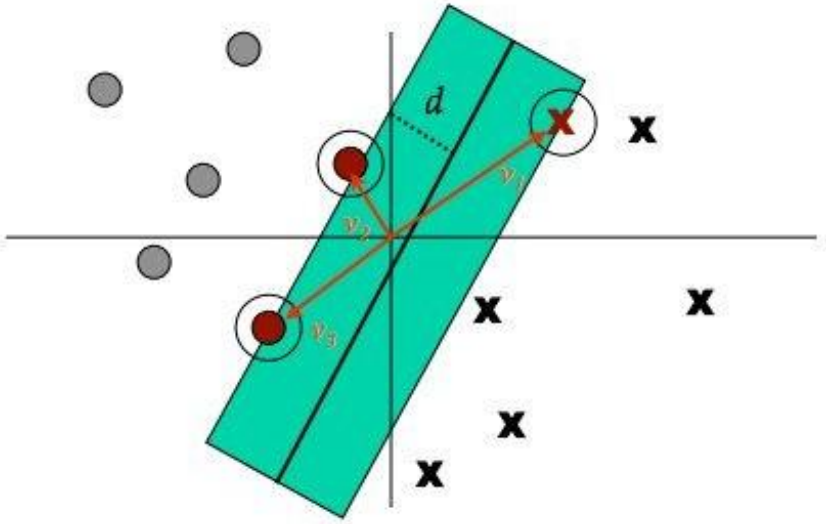
Para vermelho, $ax + by > c$, e;

Para verde, $ax + by < c$.

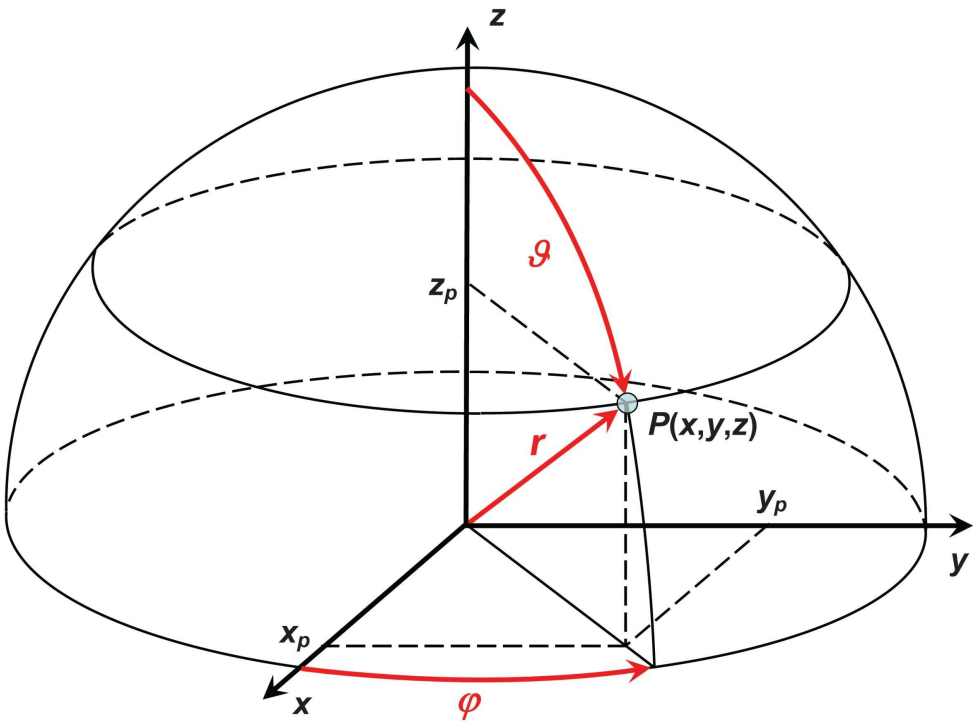
http://www.svms.org/tutorials/Berwick2003.pdf



ACHAR OS VETORES V1, V2 E V3



SVM PRESSUPÕE LINEARIDADE - PROBLEMAS (DE NOVO!):



Coordenadas polares! (Cálculo 2)

http://www.svms.org/tutorials/Berwick2003.pdf

OUTRAS TRANSFORMAÇÕES

$$K(x,y) = (x \cdot y + 1)^{p}$$

$$K(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$$

$$K(x,y) = \tanh(kx \cdot y + \delta)$$

1st is polynomial (includes x . X as special casa)

2nd is radial basis function (gaussians)

3rd is sigmoid (neural net activation function)

Obrigado

profDiogenes.Justo@fiap.com.br

SHIFT
FIAP

apêndice

Técnicas de solução: multiplicador (ou operador) de Lagrange (lagrangiano). Problema matemático de otimização:
encontrar máximos e mínimos de funções sujeitas a restrições.

LIMITE X SEM LIMITES

http://www.seer.ufrgs.br/rita/article/download/rita_v14_n2_p43-67/3543

http://www.svms.org/tutorials/Berwick2003.pdf

ESCALANDO SOLUÇÕES

LIMITE SUPERIOR X SEM LIMITES

VERTICAL

VS.

HORIZONTAL

OBS: incluindo multicore

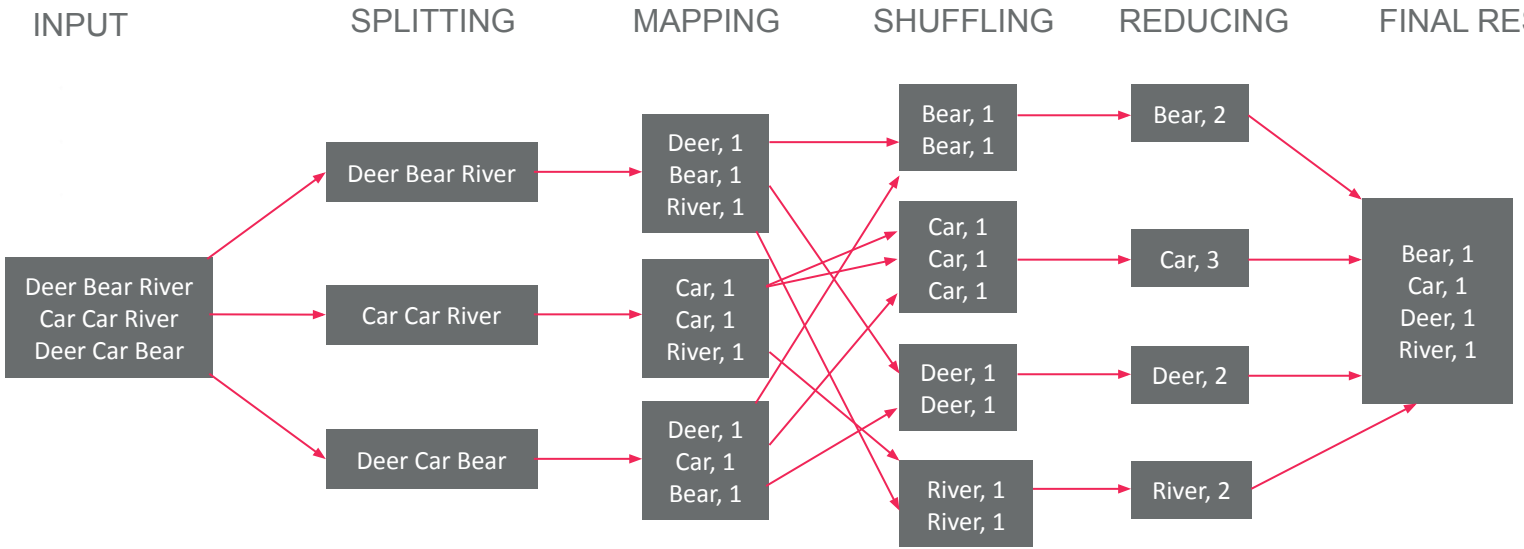
SCALE-UP LIMIT

ESCALANDO SOLUÇÕES

- Em ordem de crescimento:
- Hadoop / SPARK
- BD SQL e NoSQL em nuvem (Scale-Up e Scale-out)
- Bancos de dados SQL e NoSQL (Scale-Up)
- CSV/Desktop Processing

http://www.infomoney.com. br/bloomberg/mercados/noticia/4054301/supercomputador-para-hedge-funds-que-esta-arrasando-wall-street

THE OVERALL MapReduce WORD COUNT PROCESS



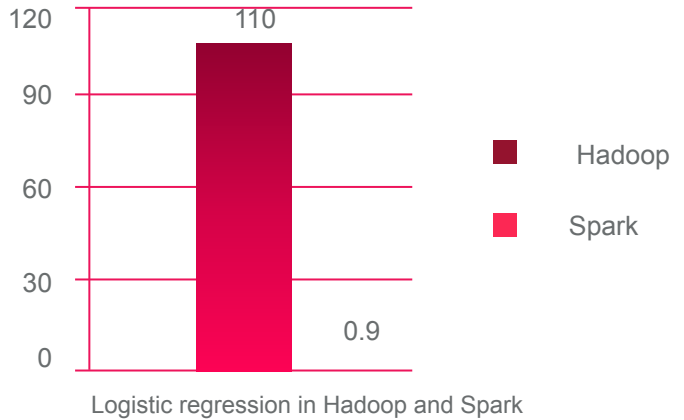
http://research.google.com/archive/mapreduce-osdi04-slides/index.html

Hadoop e Spark

SPEED

Run programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk.

Spark has an advanced DAG execution engine that supports cyclic data flow and in-memory computing.



Logistic regression in Hadoop and Spark

EASE OF USE

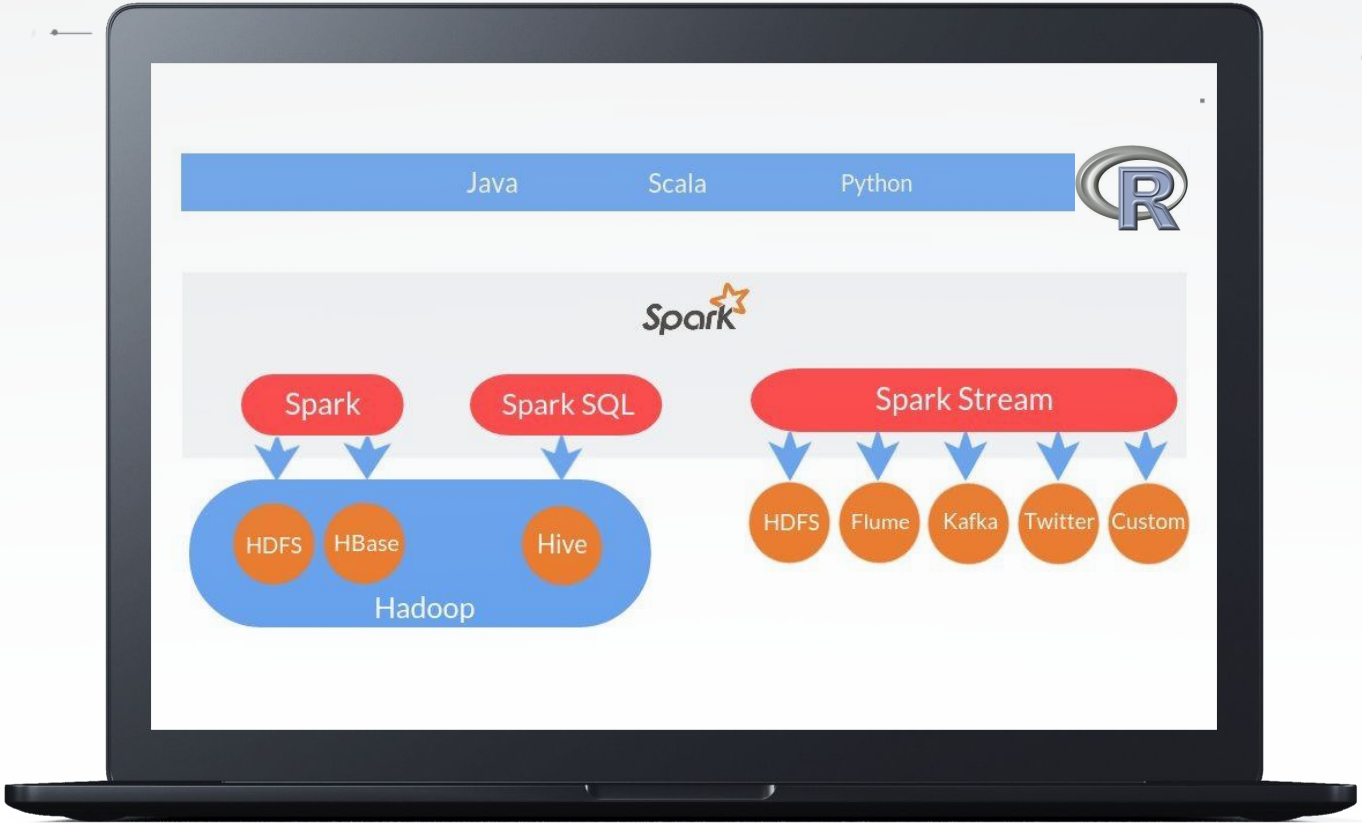
Write applications quickly in Java, Scala, Python, R.

Spark offer over 80 high-level operators that make it easy to build parallel apps. And you can use it *interactively* from the Scala, Python and R shells.

```
text_file – spark.textFile(“hdfs://…”)

text_file. flatMap(lamba line: line.split())
      .map(lambda word: (word, 1))
      .reduceByKey (lambda a, b: a+b)
```

Word count in Spark’s Python API

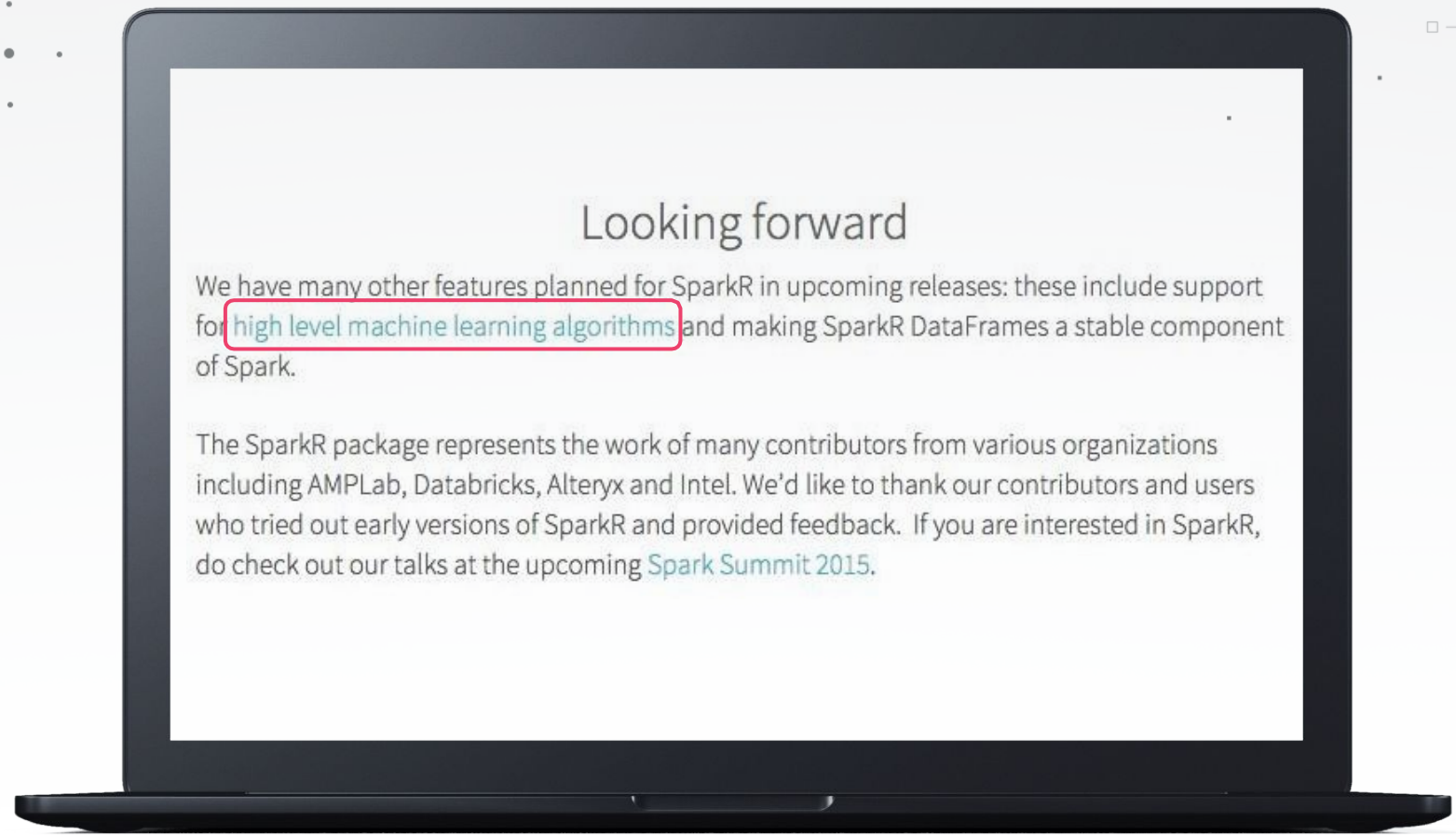
ARQUITETURA E SPARK



SPARK 1.6 (COM SPARKR)

A partir da versão 1.4, lançada em jun/2015, incorporado o SparkR

https://spark.apache.org/docs/1.6.0/sparkr.html


Looking forward
We have many other features planned for SparkR in upcoming releases: these include support for high level machine learning algorithms and making SparkR DataFrames a stable component of Spark.
The SparkR package represents the work of many contributors from various organizations including AMPLab, Databricks, Alteryx and Intel. We'd like to thank our contributors and users who tried out early versions of SparkR and provided feedback. If you are interested in SparkR, do check out our talks at the upcoming Spark Summit 2015.

CLUSTER SPARK - CUSTO CLOUD

m3.xlarge

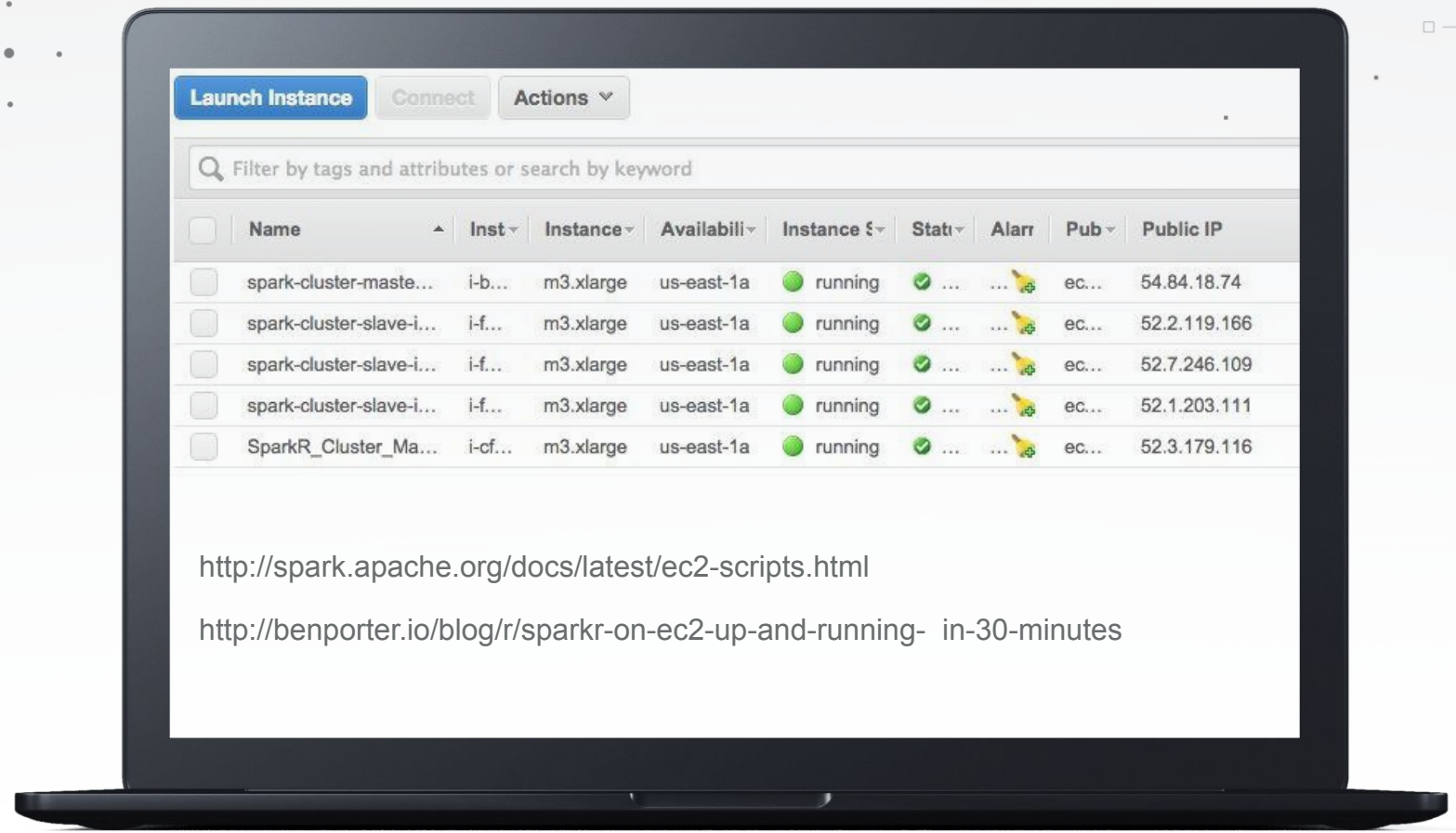
aws.amazon.com/pt/ec2/pricing/

PERÍODO DE 1 ANO					
Opção de pagamento	Inicial	Mensalmente*	Efetiva por hora**	Economia em relação ao On-Demand	On-Demand por hora
Sem taxas iniciais	$0	$138.7	$0.19	29%	$0.266 por hora
Adiantado parcial	$842	$51.1	$0.1662	38%	
Adiantado integral	$1428	$0	$0.1631	39%	

PERÍODO DE 3 ANOS					
Opção de pagamento	Inicial	Mensalmente*	Efetiva por hora**	Economia em relação ao On-Demand	On-Demand por hora
Adiantado parcial	$1345	$43.8	$0.1112	58%	$0.266 por hora
Adiantado integral	$2746	$0	$0.1045	61%	

CLUSTER SPARK - TIPO ESCOLHIDO

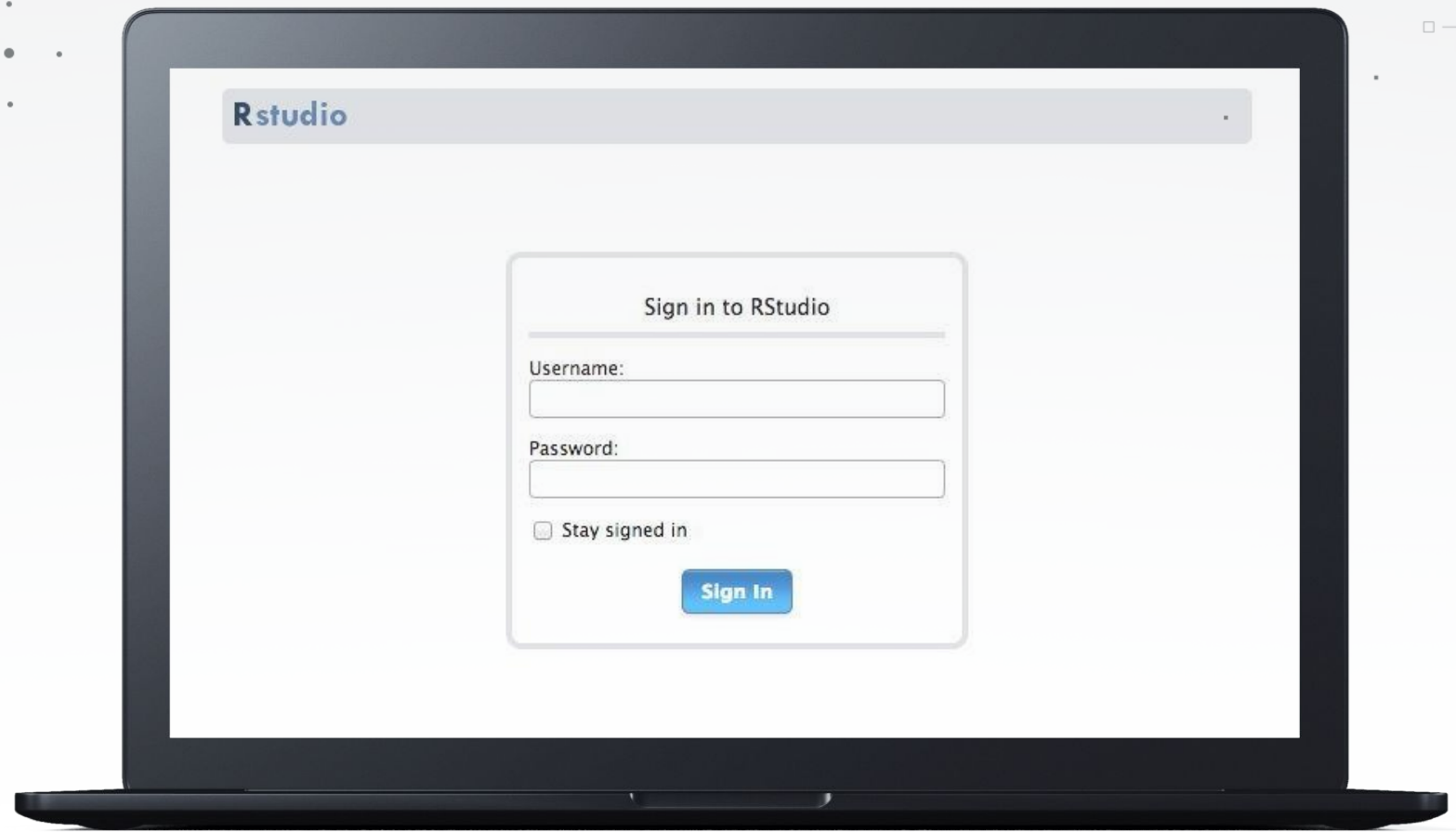
CLUSTER SparkR+RStudio no EC2



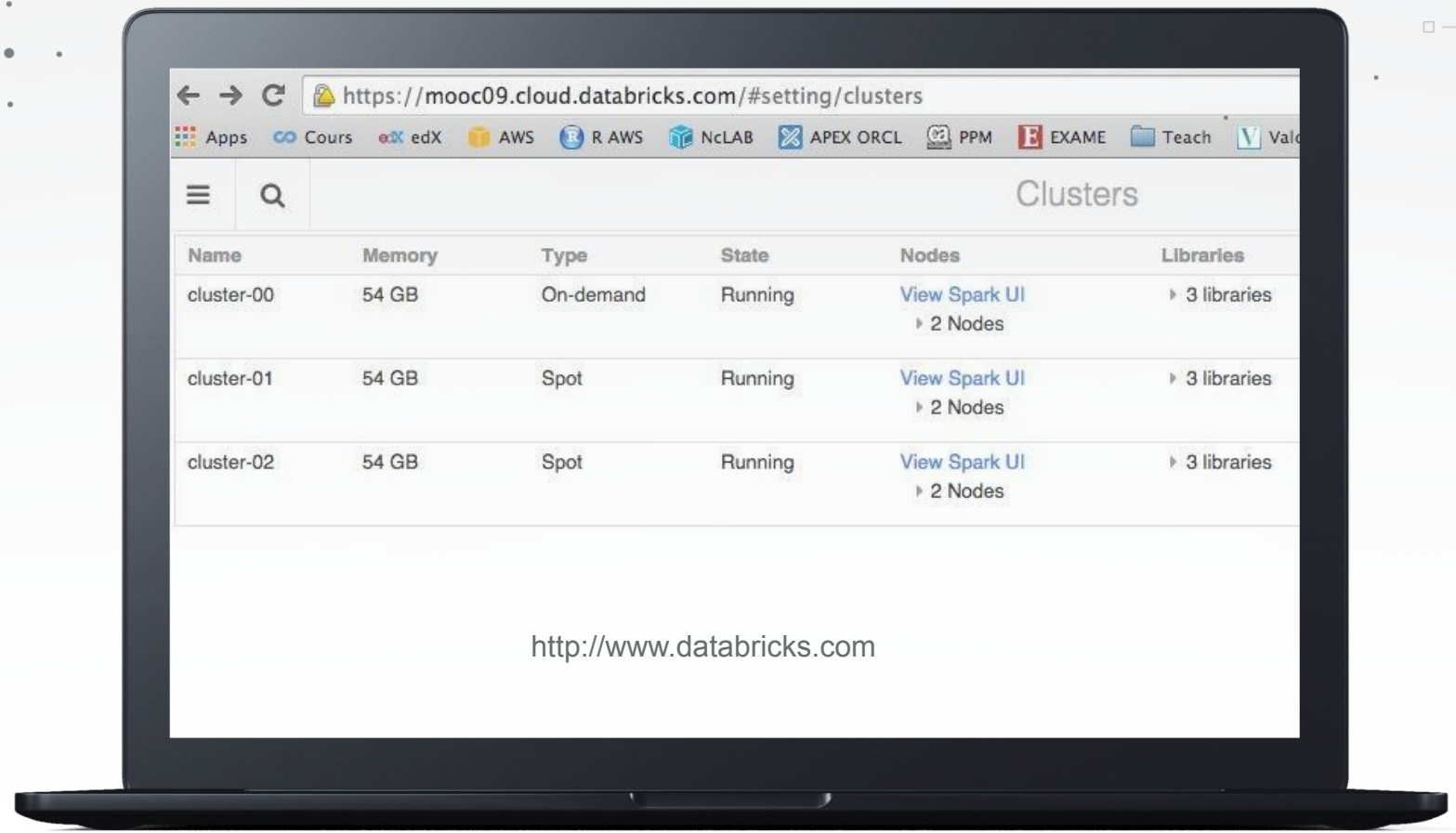
http://spark.apache.org/docs/latest/ec2-scripts.html

http://benporter.io/blog/r/sparkr-on-ec2-up-and-running- in-30-minutes

CLUSTER SparkR+RStudio no EC2

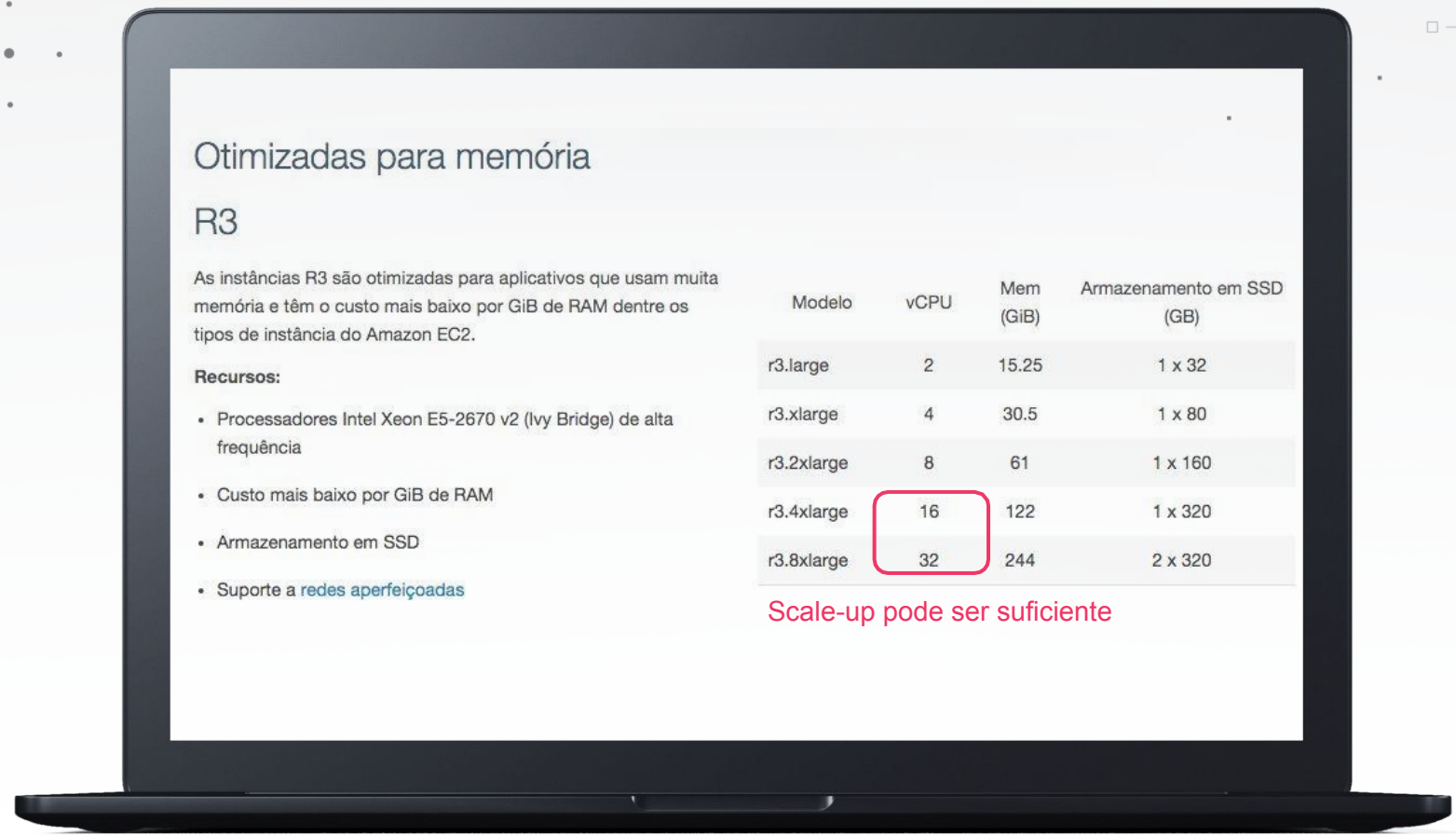


CLUSTER Spark NA DATABRICKS

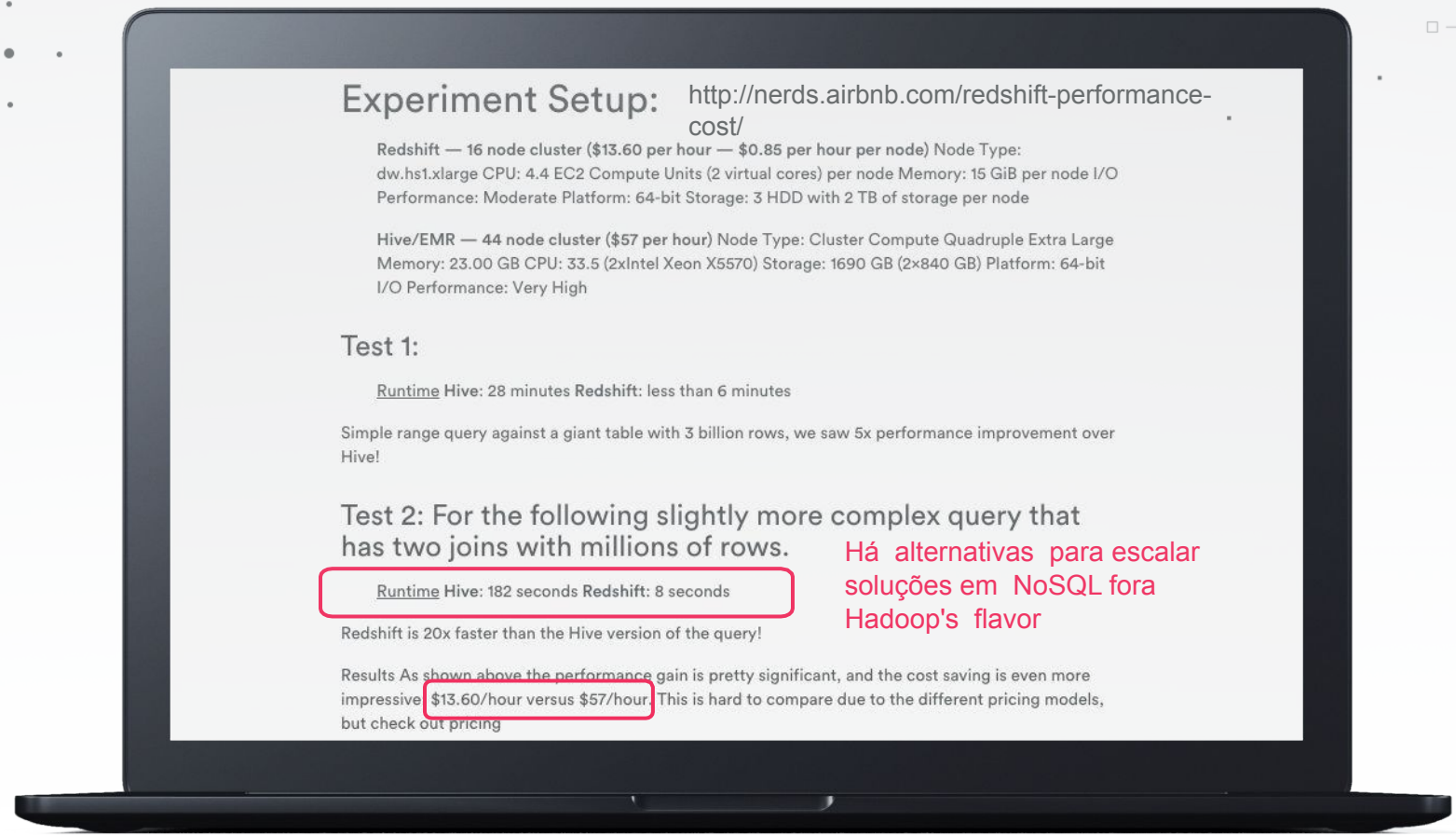


http://www.databricks.com

CLUSTER Spark NA DATABRICKS

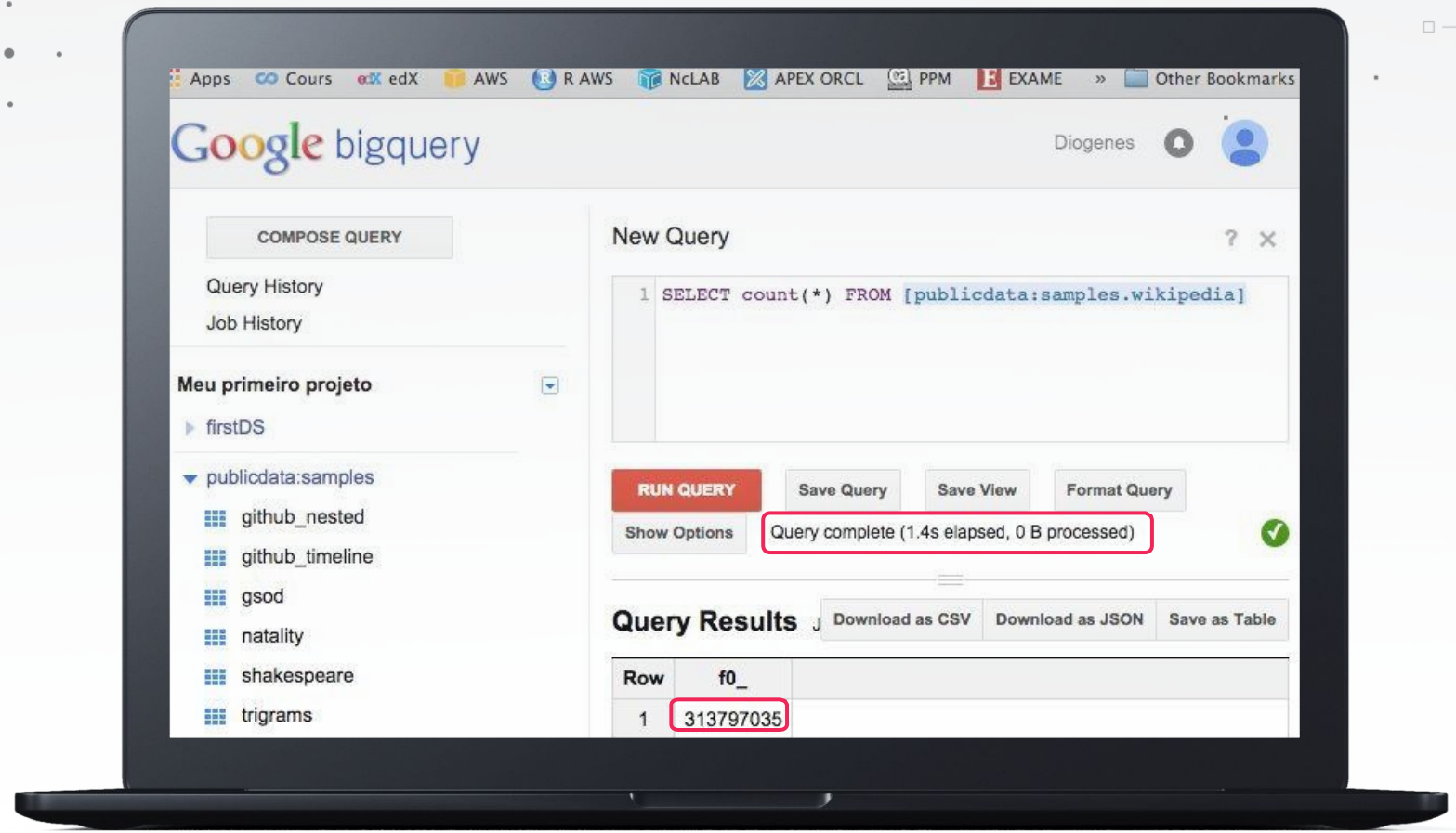


EU PRECISO DE UM CLUSTER?



Há alternativas para escalar soluções em NoSQL fora Hadoop's flavor

EU PRECISO DE UM CLUSTER?



EXERCÍCIO 5 - R SCALE-UP

CONSIDERAÇÕES PARA USO DO R EM SCALE-UP

Verificar os pacotes adequados:

- Monitorar uso de memória (pryr, mem_used())
- Avaliar tempo de processamento
- Read.csv: usar ff package
- Uso do data.table

EXERCÍCIO

Trabalhando com volumes de dados, monitore memória

```
one <- 1:10e5
print(object.size(one), units = "GB")
#Ou acesse Environment, opção grid

install.packages("pryr")
library(pryr)
mem_used()

#Remover objetos
rm(one)
```

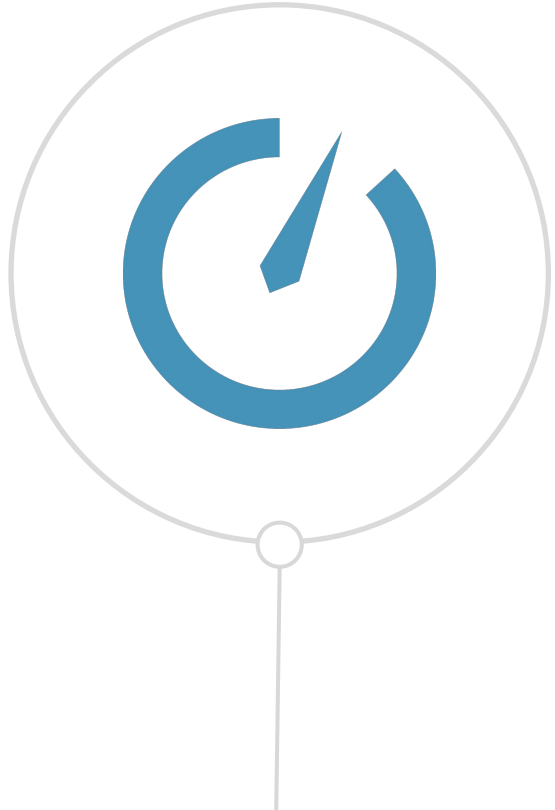
EXERCÍCIO

Monitore tempo de execução

#Opção 1: com execução
Sys.time()
# Comandos
Sys.time()

#Opçã 2: sem execução

system.time(#Comandos)



EXERCÍCIO

Carga de dados com ff package (não utiliza memória principal, deposita o data.frame em disco)

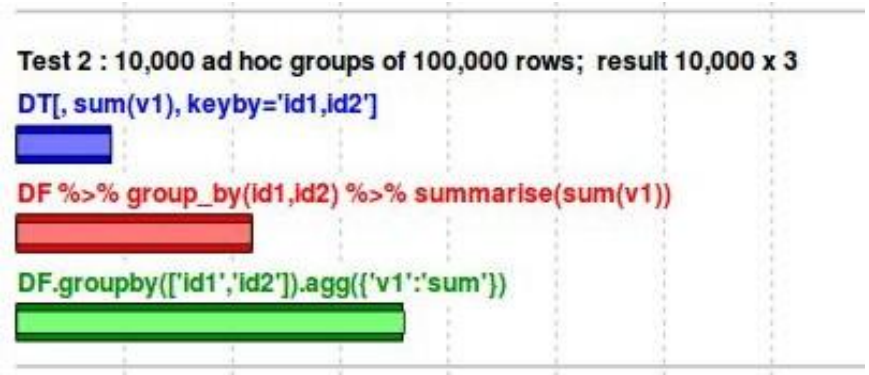
```
install.packages("ff")
library(ff)

ldf <- read.csv.ffdf(file="large.csv",

nrows=1000000,sep="#", header=TRUE, first.
rows=5000, next.rows=50000, VERBOSE=TRUE)

#next.rows divide em leituras parciais
#Verbose mostra para bloco
```

EXERCÍCIO

Uso de data.table (ao invés de data.frame)

```
install.packages("data.table")
library(data.table)

dt <- as.data.table(df)
```

https://github.com/Rdatatable/data.table/wiki/Benchmarks-%3A-Grouping



RECOMENDAÇÕES PARA ESTUDOS

- Processamento
  - Considere seriamente MapR: Hadoop (PRD) e Spark (Test) massivo?
  - Cursos MMDS (Stanford) e Intro Data Science (Bill Howe)
  - Estudar algos de redução de dimensão (PCA, SVD)
  -
  - Embasamento acadêmico ou desenvolvimento algos?
- Processo Data Science?
  - Cursos MMDS, Machine Learning (Andrew Ng), Curso Data Science (John Minning (Illinois at UC) Hopkins)