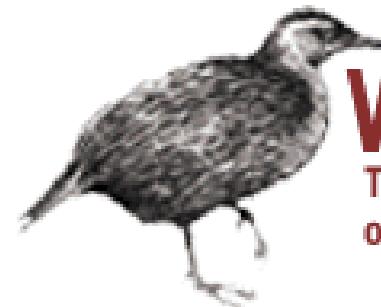


2017
PentahoDay

Curitiba . Brasil



WEKA
The University
of Waikato

Passos para o Aprendizado de Máquina com Pentaho

Prof. Marcos Vinicius Fidelis
UTFPR/UEPG

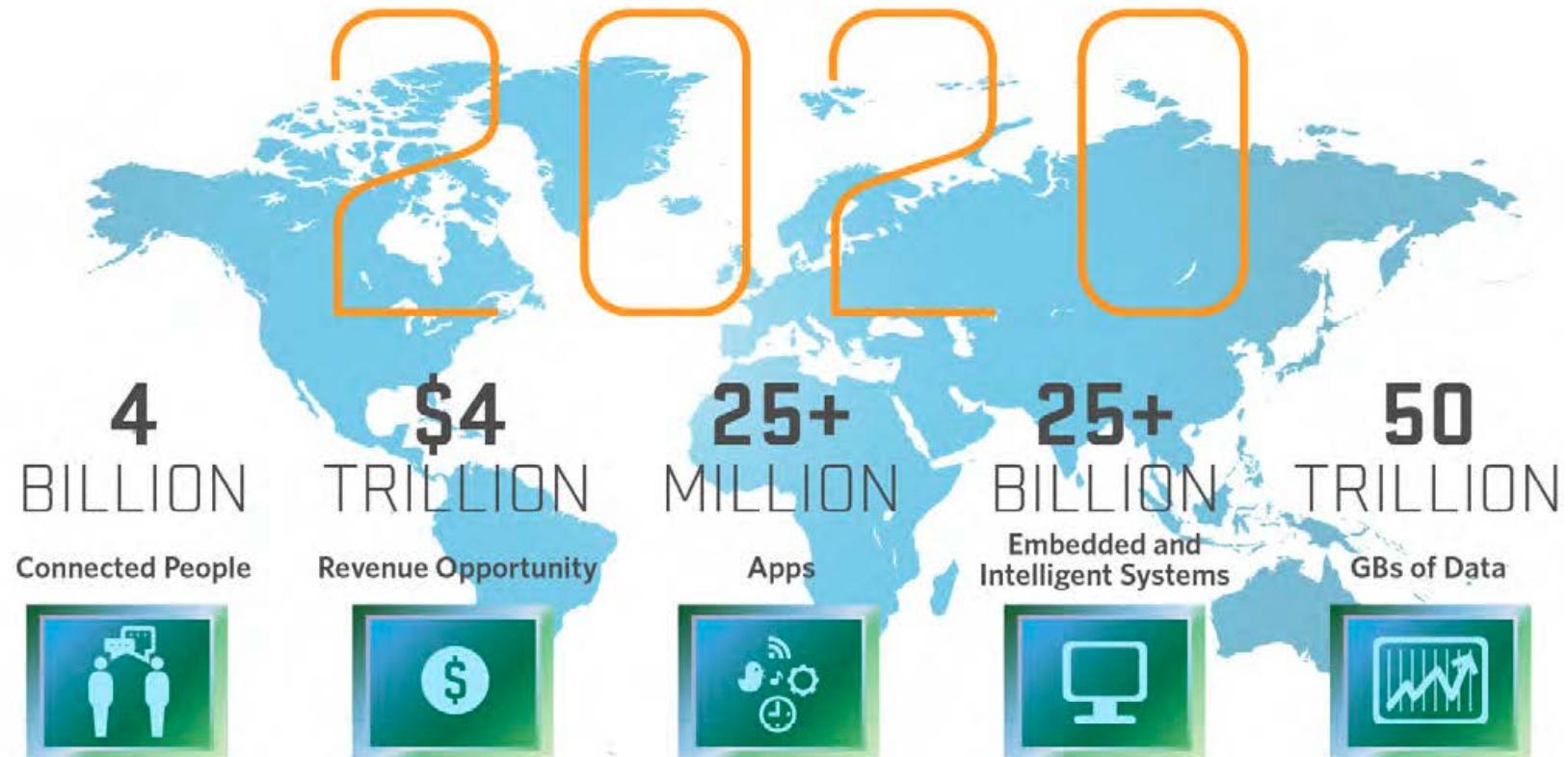
Apresentação



- Professor da UTFPR, leciona Sistemas de Apoio a Decisão e Tópicos Avançados em BD (WEKA, Pentaho e PostgreSQL).
- Analista de Informática da UEPG, administra o Sistema de Gestão Acadêmica (Grails e JasperReports).
- Atua a 25 anos com TI e com ensino de graduação.
- Entusiasta de Software Livre e Código Aberto a 7 anos.
- Membro da comunidade Pentaho Brasil e ASL.
- Palestrante em eventos como FISL, Flisol, Latinoware, FTSL, BettBrasil e PentahoDay

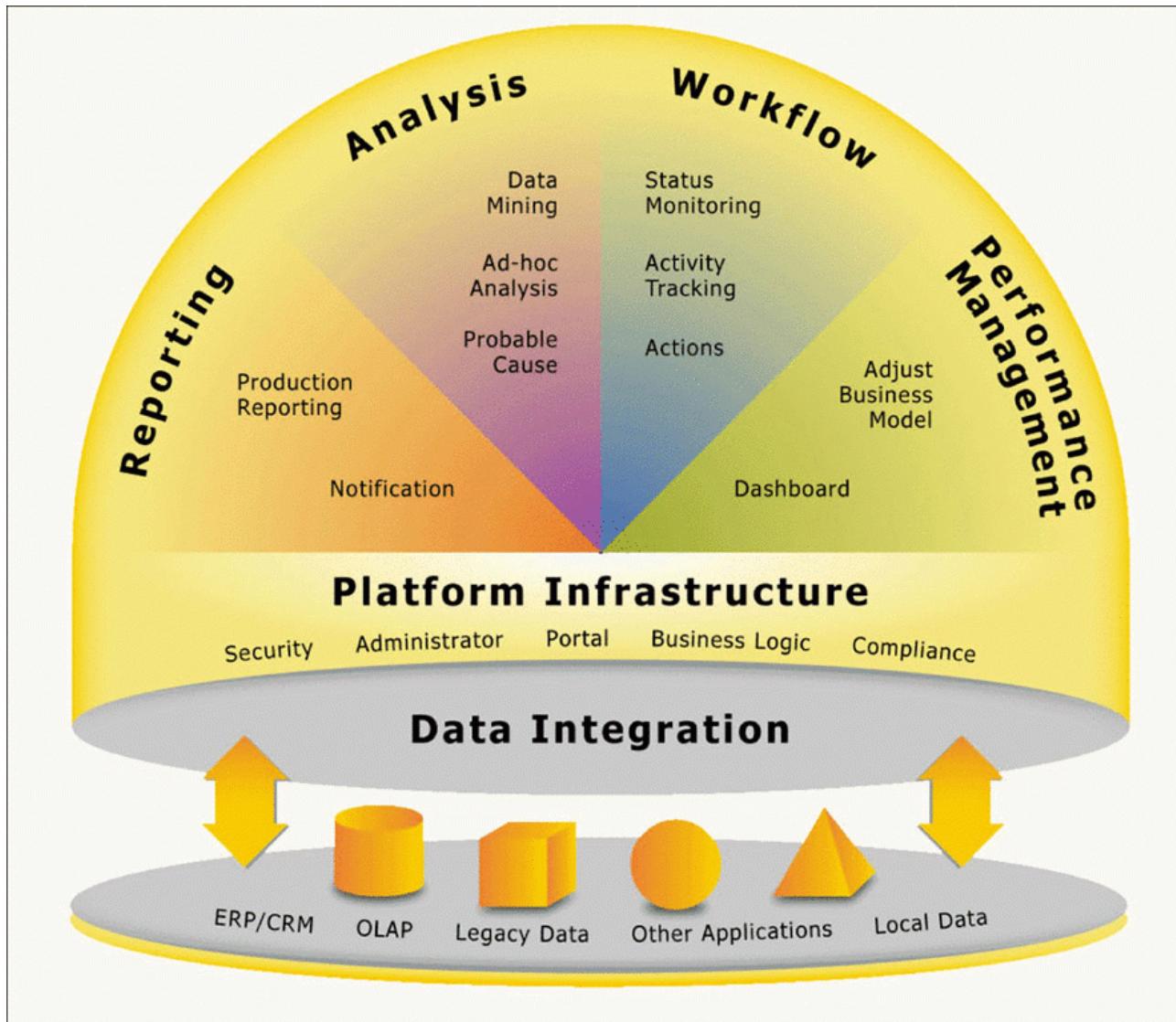


Panorama futuro...



Source: Mario Morales, IDC

Plataforma Pentaho



Por que usar Análise Preditiva?

Análise estatística e Análise de dados OLAP buscam resolver questões relacionadas ao que aconteceu historicamente.

Métodos preditivos trabalham com os mesmos dados históricos, mas tentam encontrar padrões interessantes na perspectiva de negócios.

Assume-se que estes padrões generalizam dados futuros e por isto podem ser usados para fazer previsões.

Deste modo pode-se fazer “tomar decisões” em tempo-real, por exemplo, a oferta de promoções especiais para clientes específicos.



Pentaho Data Mining (a.k.a WEKA)

- Incorpora várias técnicas de ML em um software chamado WEKA
- Waikato Ambiente para a análise de conhecimento (*Waikato Environment for Knowledge Analysis*)
- Com ele, um especialista em um área de conhecimento em particular é capaz de usar ML para derivar conhecimento útil a partir de bancos de dados que são demasiado grandes para serem analisados manualmente
- Os usuários do WEKA são pesquisadores da ML e cientistas industriais
- É também amplamente utilizado para o ensino
- Escrito em Java e distribuído sob a Licença Pública Geral (GNU)



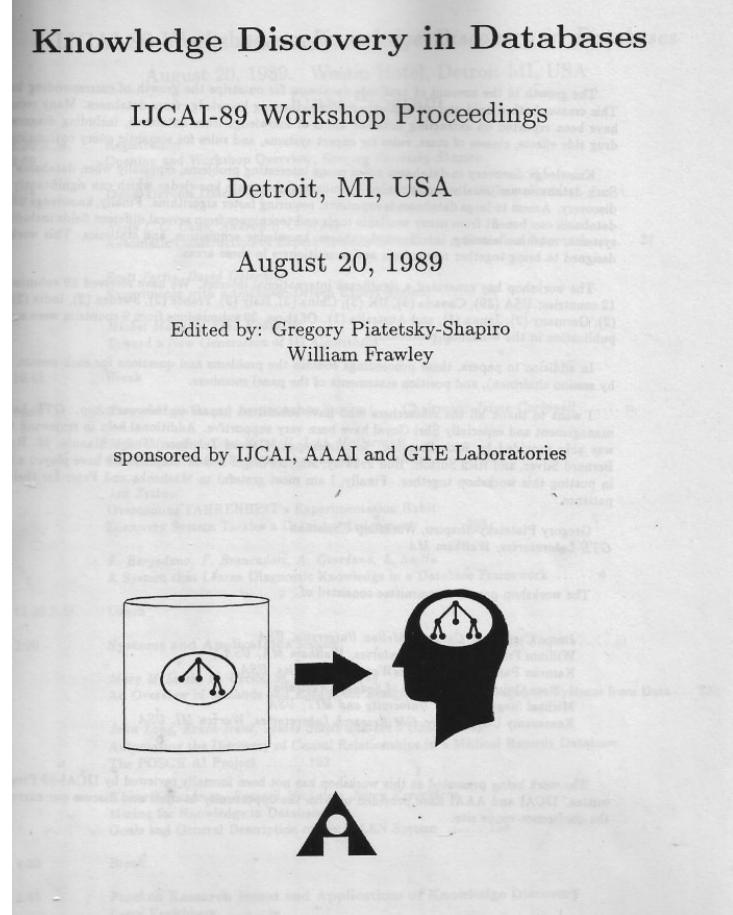
Histórico

- Principais versões do Weka
 - Weka 3.4 - ramo estável que foi criado em 2003 para corresponder com o que está descrito na segunda edição do livro Witten e Frank Data Mining (publicado 2005) . Recebe apenas correções de bugs.
 - Weka 3.6 - ramo estável que foi criado em meados de 2008 para corresponder com o que está descrito na 3^a edição do Witten, Frank e Hall - livro Data Mining (publicado em janeiro de 2011) . Recebe apenas correções de bugs.
 - Weka 3.8 – Última versão estável.
 - Weka 3.9 – versão de desenvolvimento . Esta é uma continuação da versão 3.8 Recebe correções de bugs e novos recursos.
- Anteriormente
 - 1992 – submissão do projeto ao governo de NZ (Ian Witten)
 - 1993 – aprovado pelo governo
 - 1994 – Primeira versão (principalmente em C)
 - 1996 – Primeira versão pública – WEKA 2.1
 - 1997 – Convertido para Java
 - 1998 – WEKA 3 (completamente Java)
 - 2006 – O projeto foi incorporado ao Pentaho

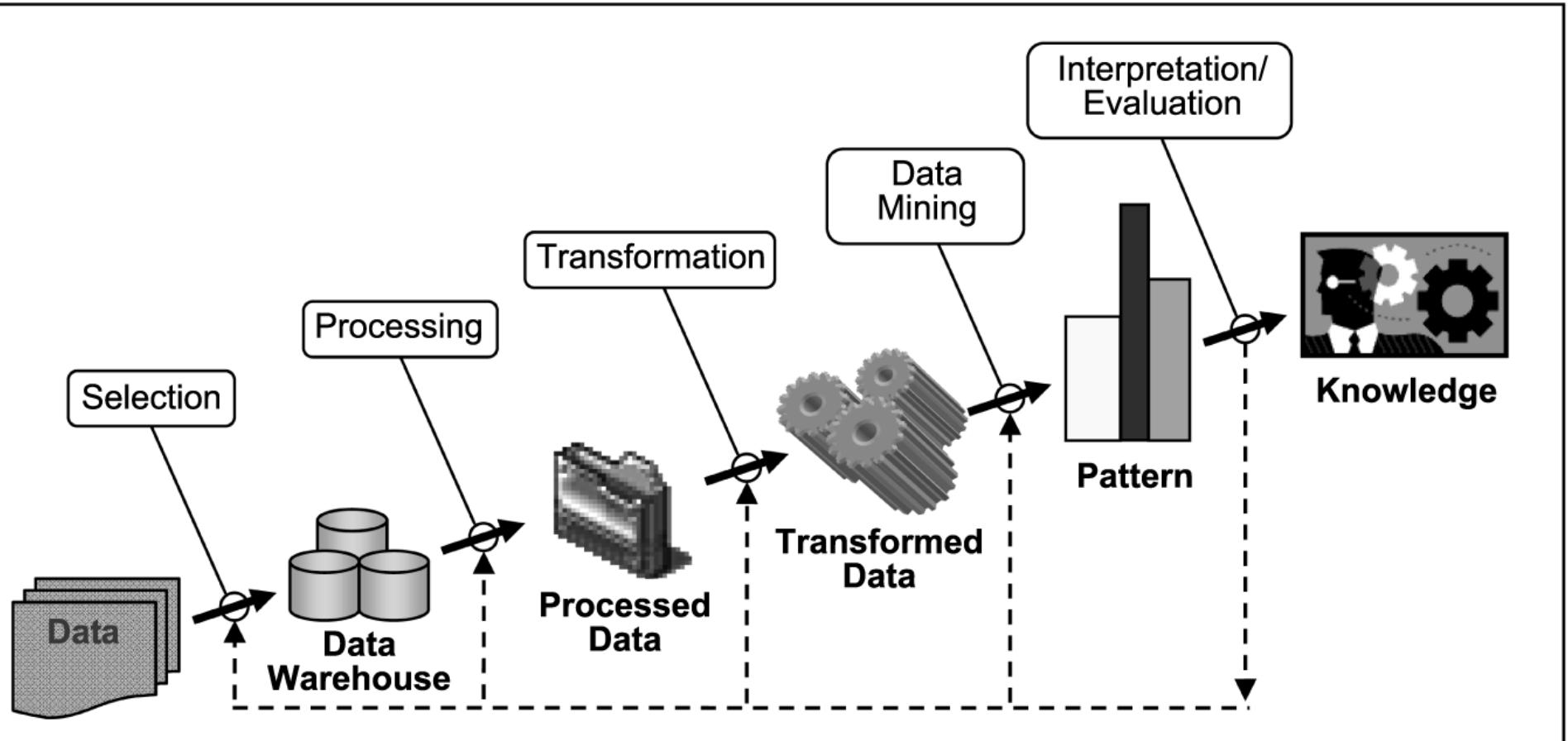


Data Mining: origens

- Data Mining é apenas uma etapa de um processo de KDD
- 1989 - Gregory Piatetsky-Shapiro organiza e coordena o primeiro Workshop Knowledge Discovery in Databases (KDD).
- 1995 – tornou-se ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)



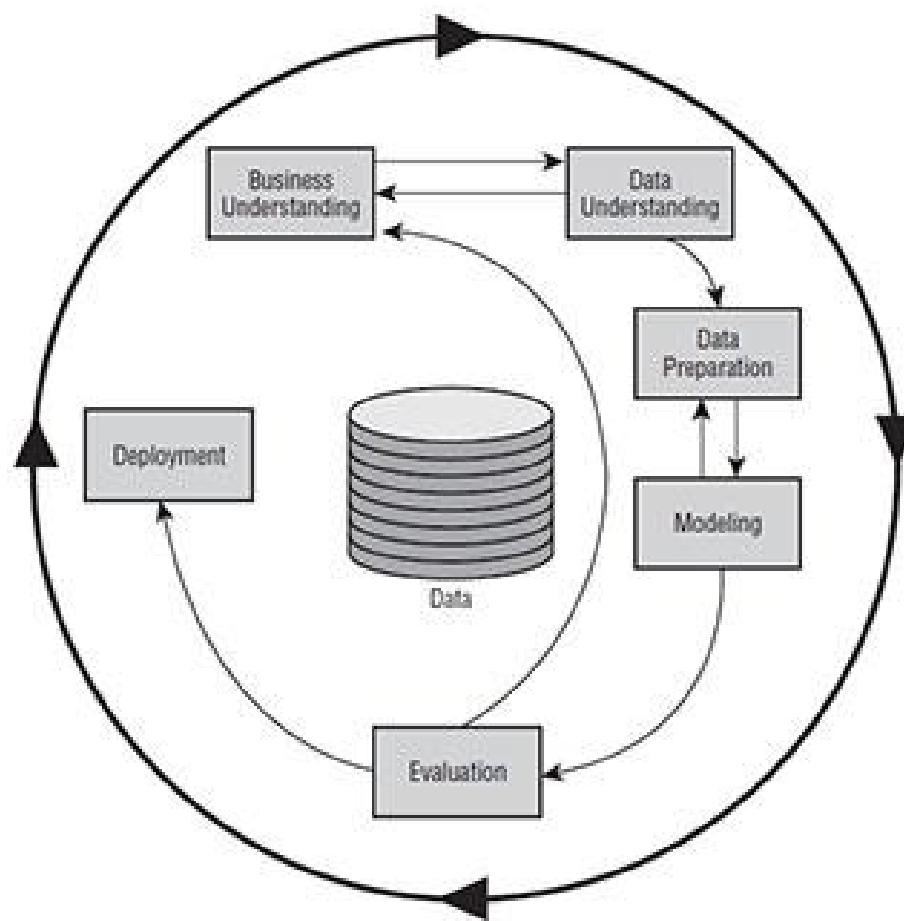
Fases do Processo de Descoberta de Conhecimento (Knowledge Discovery in Databases)



Cross Industry Standard Process for Data Mining

- *Cross Industry Standard Process for Data Mining*
- Processo Padrão Inter-Indústrias para Mineração de Dados
- É um modelo de processo de mineração de dados que descreve abordagens comumente usadas por especialistas em mineração de dados para atacar problemas.

O Processo CRISP-DM

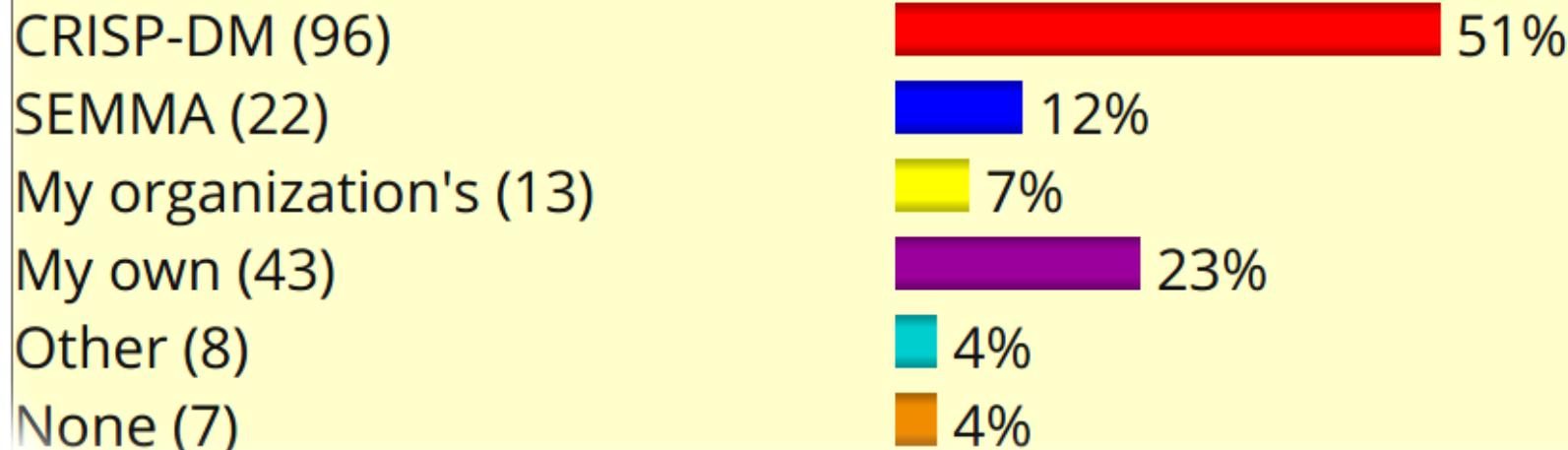


Fases

- Entender o Negócio: foca em entender o objetivo do projeto a partir de uma perspectiva de negócios, definindo um plano preliminar para atingir os objetivos.
- Entender os Dados: recolhimento de dados e inicio de atividades para familiarização com os dados, identificando problemas ou conjuntos interessantes.
- Preparação dos Dados: construção do conjunto de dados final a partir dos dados iniciais. Normalmente ocorre várias vezes no processo.
- Modelagem: várias técnicas de modelagem são aplicadas, e seus parâmetros calibrados para otimização. Assim, é comum retornar à Preparação dos Dados durante essa fase.
- Avaliação: é construído um modelo que parece ter grande qualidade de uma perspectiva de análise de dados. No entanto, é necessário verificar se o modelo atinge os objetivos do negócio.
- Implantação: o conhecimento adquirido pelo modelo é organizado e apresentado de uma maneira que o cliente possa utilizar.

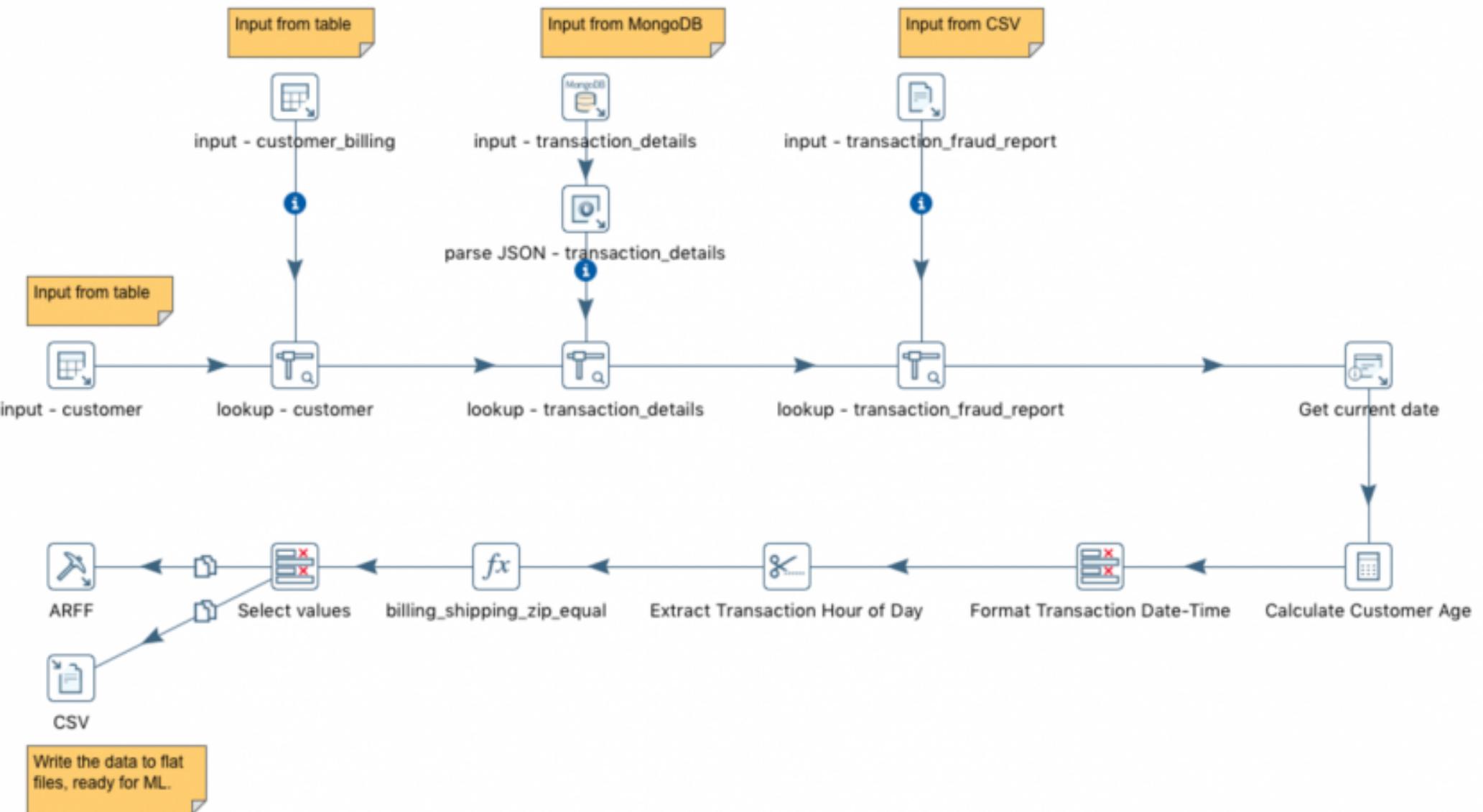
Poll

What main methodology are you using for data mining? [189 votes total]



PDI pode ser sua ferramenta de pré-processamento de dados

- Com o PDI pode-se combinar fontes de dados diferentes
- Dados do cliente são associados a várias tabelas de BD relacional e mesclados com dados transacionais do MongoDB e ocorrências de fraude de um arquivo CSV
- São derivados campos adicionais que podem ser úteis para modelagem preditiva
 - Idade
 - Hora do dia da compra
 - Flag de endereços



- Uma rede de varejo quer reduzir perdas devido a ordens com uso fraudulento de cartões de crédito.
- Detalhes básicos são armazenados em um BDR. Encomendas no MongoDB. Relatório com histórico de fraude em CSV.

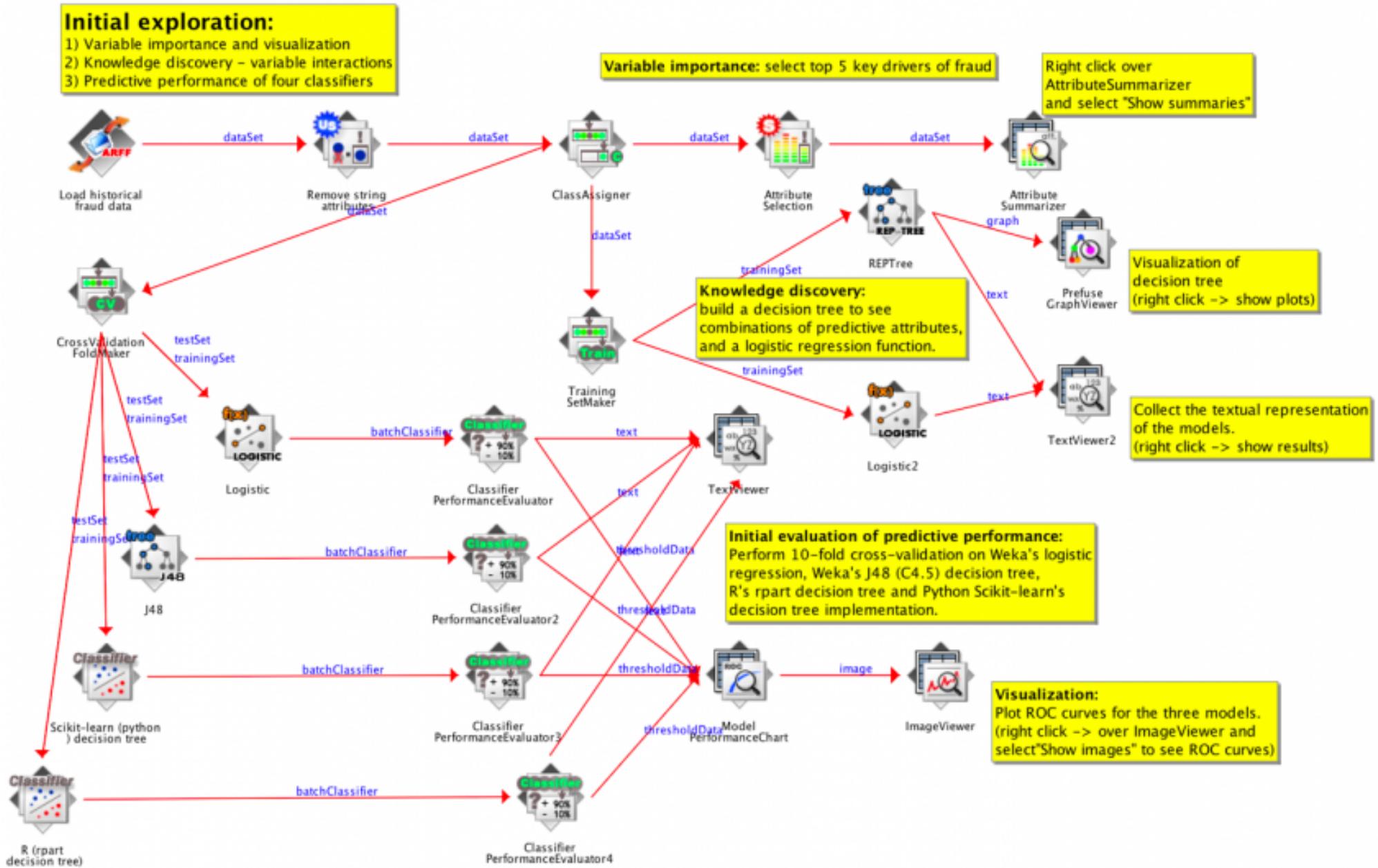
Avaliando os dados combinados pelo PDI

- Gera-se 100.000 exemplos(linhas)
- Dos atributos pode-se descartar campos como nome do cliente, ID, e-mail, num_tel e endereços físicos
- Atributos que identificam unicamente um exemplo são descartados

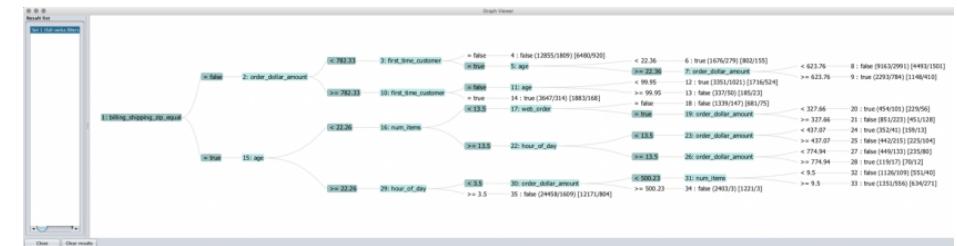
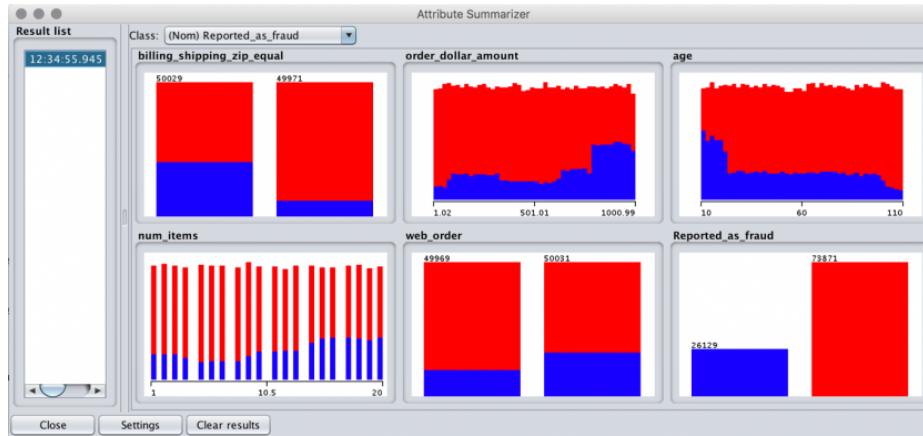
Papel do PDI

- Remover o trabalho repetitivo da preparação de dados
- Gerar dados prontos para o trabalho do Cientista de Dados realizar sua modelagem
- Trabalhar integrado com bibliotecas populares como R, Python, WEKA e Spark Mlib
- Desta forma os membros da equipe podem trabalhar em diferentes ambientes mas integrados.
- Sem ditar a ferramenta preditiva.

Modelagem em um processo típico de ML, para exploração inicial, projetado no Knowledge Flow



Modelagem – avaliações preliminares



- Principais variáveis análise de fraudes
- Interação de variáveis
- 4 diferentes algoritmos de classificação supervisionada



Logistic regression



J48 (C4.5 release 8)



Scikit-learn (python) decision tree



R (rpart decision tree)

No Knowledge Flow pode-se capturar métricas para comparar desempenho dos classificadores

Result list

Text

```
12:35:30.123 - R (rpart decis
12:35:34.073 - J48
12:35:34.191 - Scikit-learn (
12:35:34.907 - Logistic

Text Viewer
```

12:35:30.123 - R (rpart decis
12:35:34.073 - J48
12:35:34.191 - Scikit-learn (
12:35:34.907 - Logistic

Result list

Text

```
Text
==== Evaluation result ===
Scheme: J48
Options: -C 0.25 -M 2
Relation: full-weka.filters.unsupervised.attribute.Remove-R15-last_weka.datagenerators.classifiers.clas
```

Correctly Classified Instances 97335 97.335 %
Incorrectly Classified Instances 2665 2.665 %
Kappa statistic 0.9307
Mean absolute error 0.0403
Root mean squared error 0.1484
Relative absolute error 10.4451 %
Root relative squared error 33.7894 %
Total Number of Instances 100000

==== Detailed Accuracy By Class ====
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.942 0.016 0.955 0.942 0.949 0.931 0.989 0.978 true
0.984 0.058 0.980 0.984 0.982 0.931 0.989 0.994 false
Weighted Avg. 0.973 0.047 0.973 0.973 0.973 0.931 0.989 0.990

==== Confusion Matrix ====
a b <-- classified as
24622 1507 | a = true
1158 72713 | b = false

Close Settings Clear results

Image list

Image

ROC curves

The plot shows the performance of four classifiers. The Y-axis is 'True Positive Rate' ranging from 0.0 to 1.0. The X-axis is 'False Positive Rate' ranging from 0.0 to 1.0. The legend indicates:
R (part decision tree) : MLRClassifier (class: true) (Red line)
J48 (class: true) (Blue line)
Scikit-learn (python) decision tree : ScikitLearnClassifier (class: true) (Green line)
Logistic (class: true) (Yellow line)

True Positive Rate

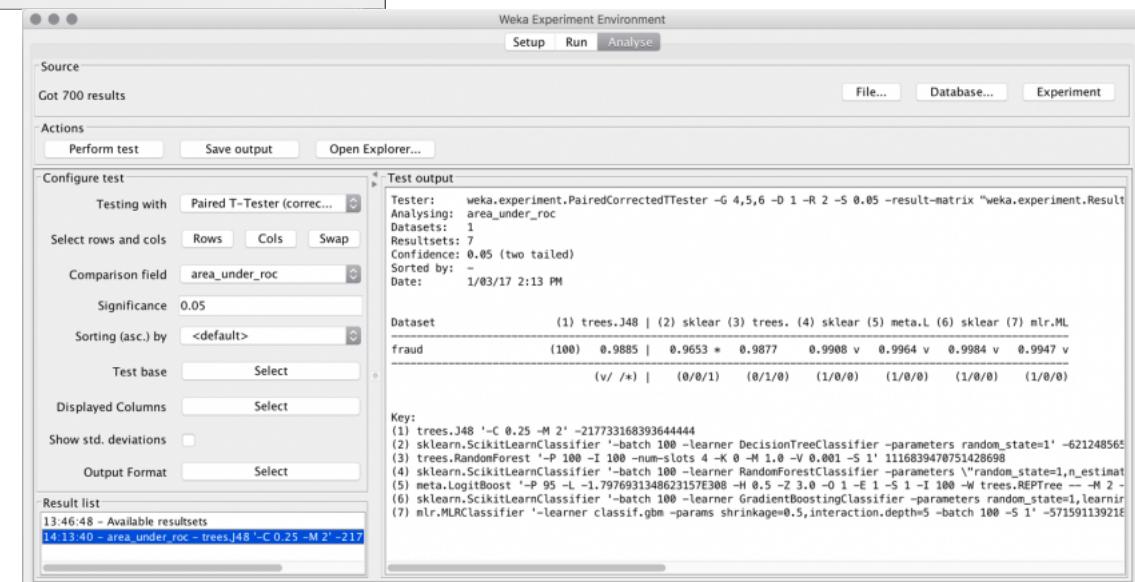
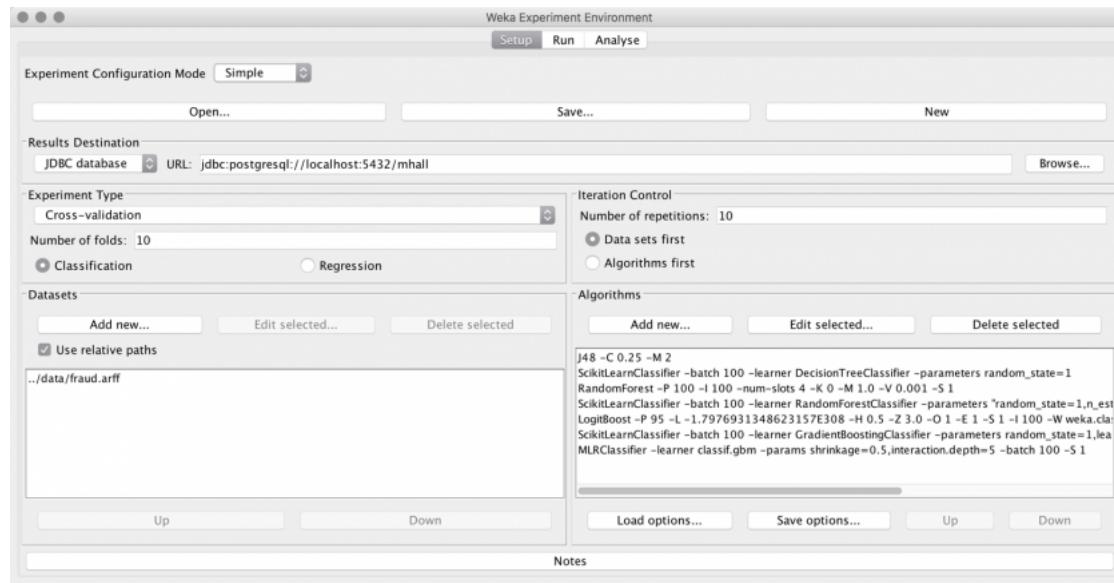
False Positive Rate

R (part decision tree) : MLRClassifier (class: true) — J48 (class: true)
Scikit-learn (python) decision tree : ScikitLearnClassifier (class: true) — Logistic (class: true)

O que tenho disponível?

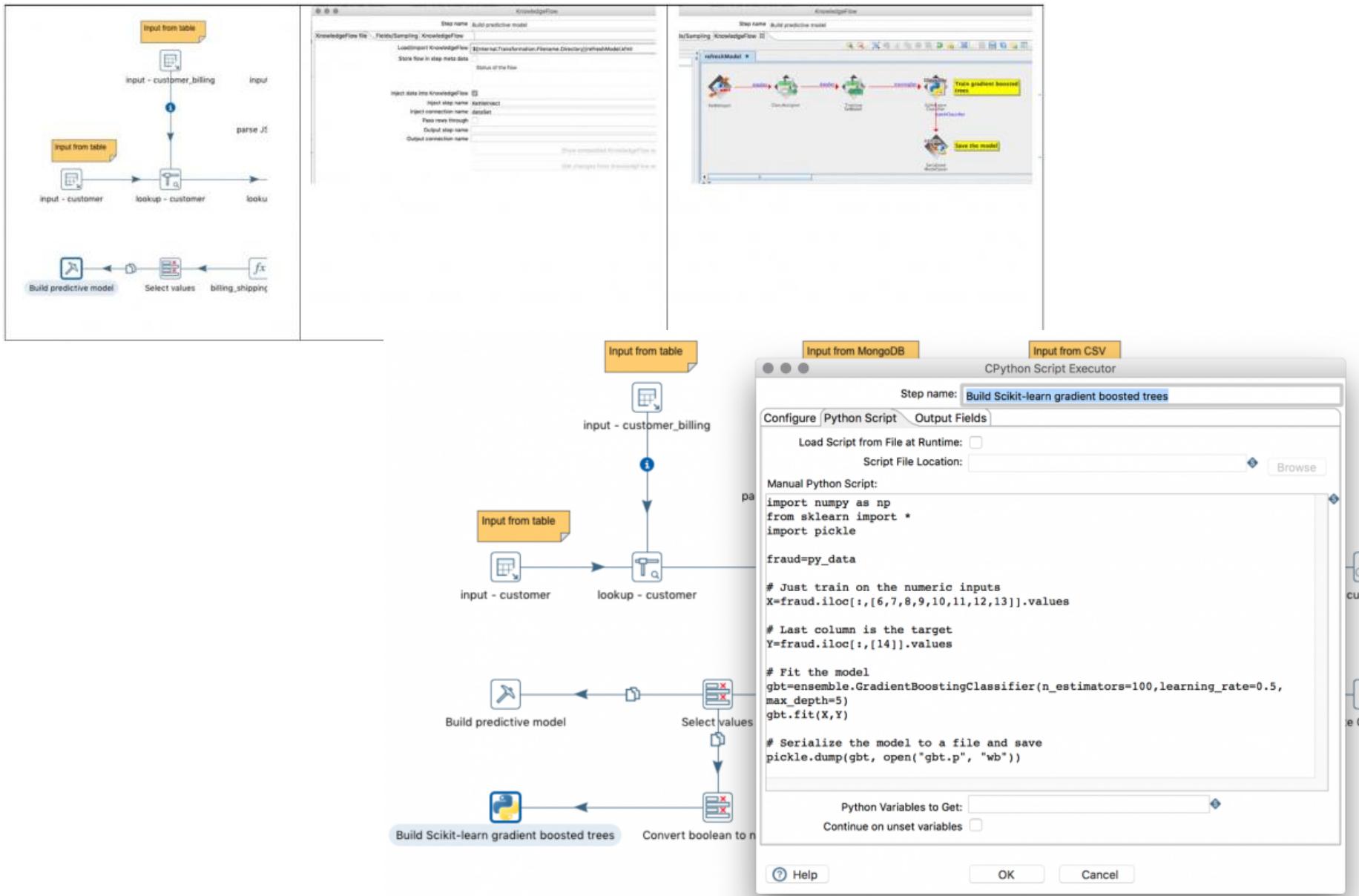
- 100+ algoritmos de classificação
- 75 para pré-processamento de dados
- 25 para apoiar o processo de Seleção de Atributos
- 20 para agrupamento, regras de associação, etc
- Atualmente possui um gerenciador de pacotes que permite adicionar novos algoritmos e ferramentas

Testar algoritmos de ML leva tempo, utilize o Experimenter



"*" e "v" indicarão se o algoritmo tem desempenho significantemente melhor ou pior do que um teste

E ainda atualizar dinamicamente seus modelos preditivos

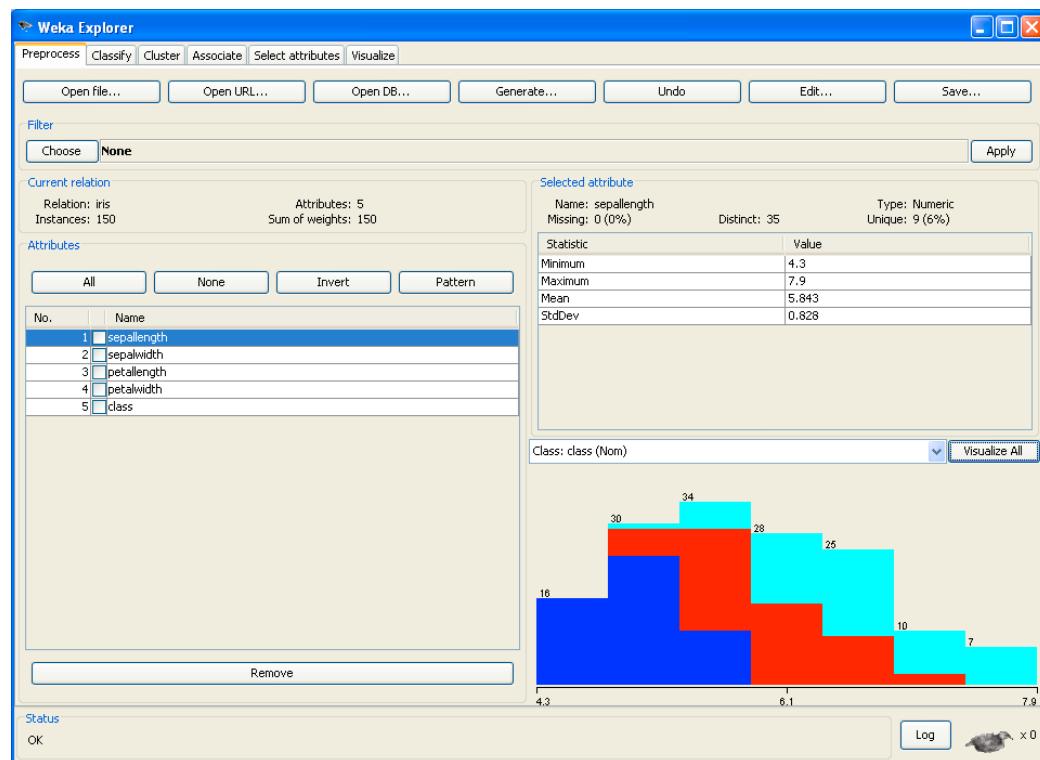


Data Integration plugins

- O *Weka Scoring Plugin*: é uma ferramenta que permite que modelos de agrupamento/classificação criados com WEKA sejam usados para pontuar (score) novos dados como parte de uma transformação no Kettle. "Scoring" significa simplesmente adicionar uma predição a uma nova linha de dados. O Weka scoring pode usar todos os tipos de classificadores e agrupadores que podem ser construídos no WEKA.
- O *ARFF Output Plugin* é uma ferramenta que permite gerar ua saída a partir do Kettle para o formato ARFF (*Attribute Relation File Format*). O formato ARFF é essencialmente o mesmo que o formato CSV, exceto pela adição de metadados dos atributos no cabeçalho.

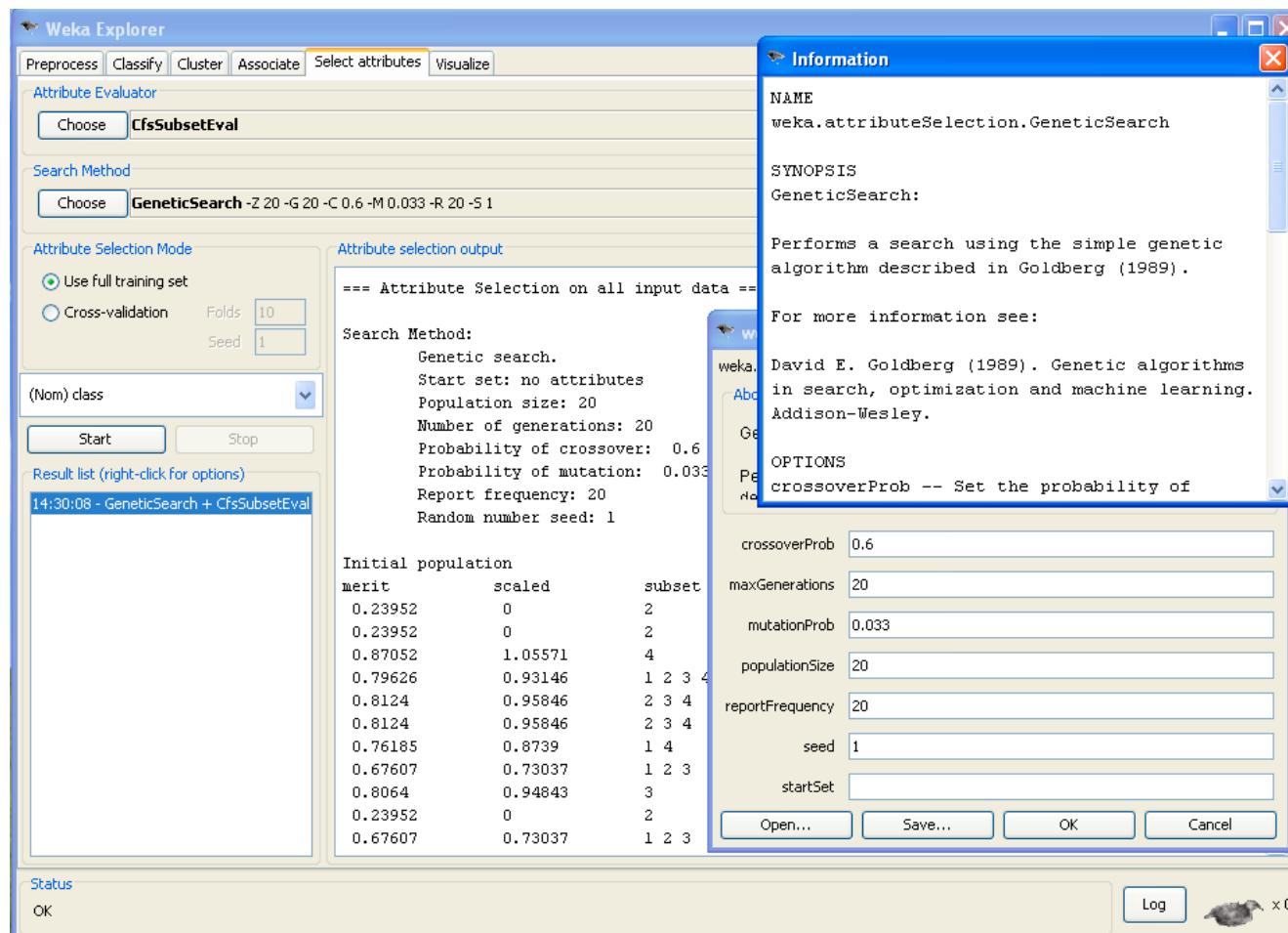
WEKA Explorer

- O painel de pré-processamento tem facilidades para importar dados de BD, CSV, etc.
- Aplicação de filtros para transformação de dados (por exemplo transformar atributos numéricos em categóricos e vice-versa)
- Permite excluir exemplos/atributos utilizando critérios



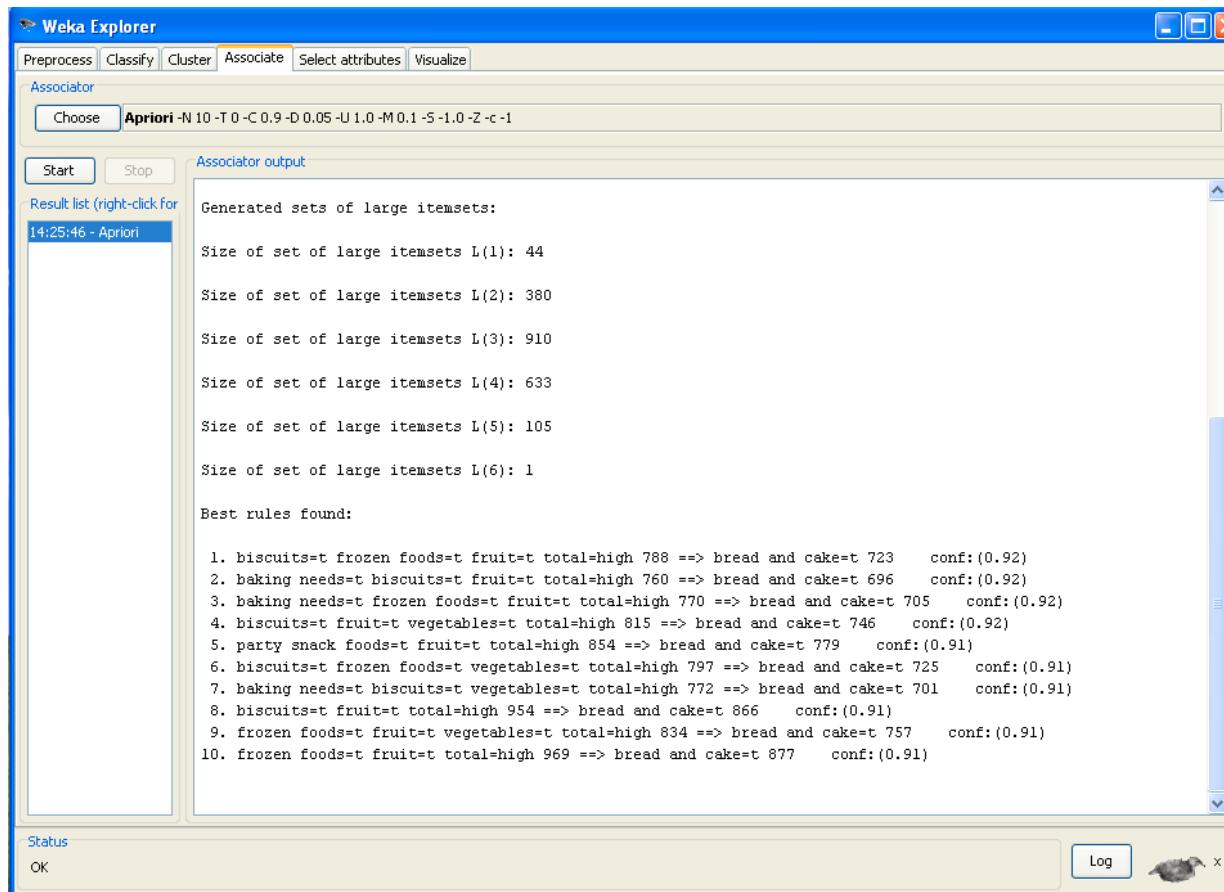
Seleção de Atributos

- São algoritmos que permitem identificar os atributos mais preditivos no *dataset*.

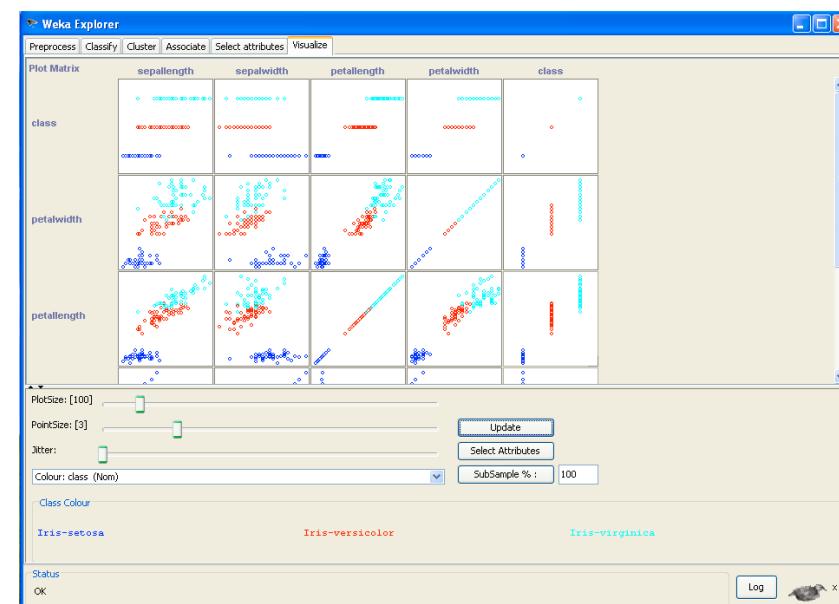
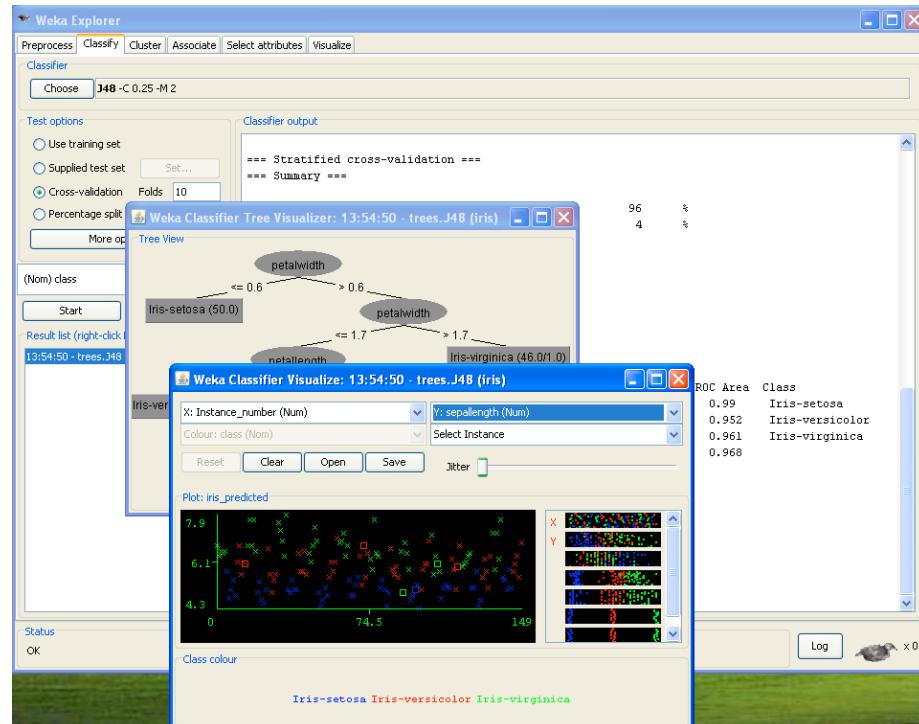


Tarefa de Associação

- Algoritmos que tentam identificar todos os interrelacionamentos importantes entre atributos nos dados.



Painéis de visualização



Knowledge Flow

The image displays two screenshots of the Weka KnowledgeFlow Environment interface, illustrating the workflow for model comparison and evaluation.

Top Screenshot: A detailed view of a KnowledgeFlow process titled "modelComparison2". The process starts with an "ArffLoader" node reading an "ARFF" file named "german_credit". The output goes to a "ClassValuePicker" node, which then splits the data into "trainingSet" and "testSet". These sets are fed into two parallel classifier nodes: "J48" and "Logistic". The outputs from both classifiers are combined via a "batchClassifier" node. The resulting predictions are evaluated by a "ClassifierPerformanceEvaluator" node, which also receives "thresholdData" from a "ModelPerformanceChart" node. The "ModelPerformanceChart" node plots the "False Positive Rate (Num)" against the "True Positive Rate (Num)" for the "german_credit" dataset. The chart shows a curve starting at (0.0014, 0.0033) and rising towards (1, 1). A legend indicates "J48 (class: good)" and "Logistic (class: good)".

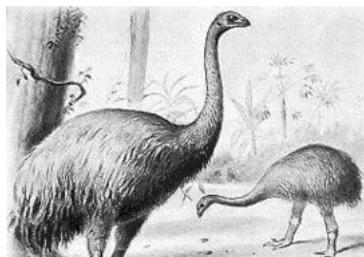
Bottom Screenshot: A broader view of the "modelComparison2" process. It includes additional nodes and annotations:

- Annotations:**
 - "Loads the closed opps data": Points to the "ArffLoader" node.
 - "Set the class attribute": Points to the "ClassAssigner" node.
 - "Choose the 'positive' class label": Points to the "ClassValuePicker" node.
 - "Perform a 10-fold cross-validation": Points to the "CrossValidationFoldMaker" node.
 - "J4p - RIPPER rules": Points to the "J4p" classifier node.
 - "Evaluate J4p": Points to the "ClassifierPerformanceEvaluator" node for J4p.
 - "Display ROC curves": Points to the "ModelPerformanceChart" node.
 - "Evaluate LogitBoost": Points to the "ClassifierPerformanceEvaluator" node for LogitBoost.
 - "Accuracy summary": Points to the "TextViewer" node.
 - "Cost/benefit analysis": Points to the "CostBenefitAnalysis" and "Cost/benefit analysis" nodes.
- Nodes:** ArffLoader, ClassValuePicker, ClassAssigner, CrossValidationFoldMaker, J4p, LogitBoost, ClassifierPerformanceEvaluator, ModelPerformanceChart, TextViewer, CostBenefitAnalysis.
- Table (Status Log):**

Component	Parameters	Time	Status
[KnowledgeFlow]		0:2:12	Flow loaded.
ArffLoader		0:0:2	Finished.
CrossValidationFoldMaker	-C 0.25-M 2	-	Finished.
J4p	-R 1.0E-8-M -1	-	Finished.
ClassifierPerformanceEvaluator		0:0:1	Finished.
ClassifierPerformanceEvaluator		0:0:2	Finished.

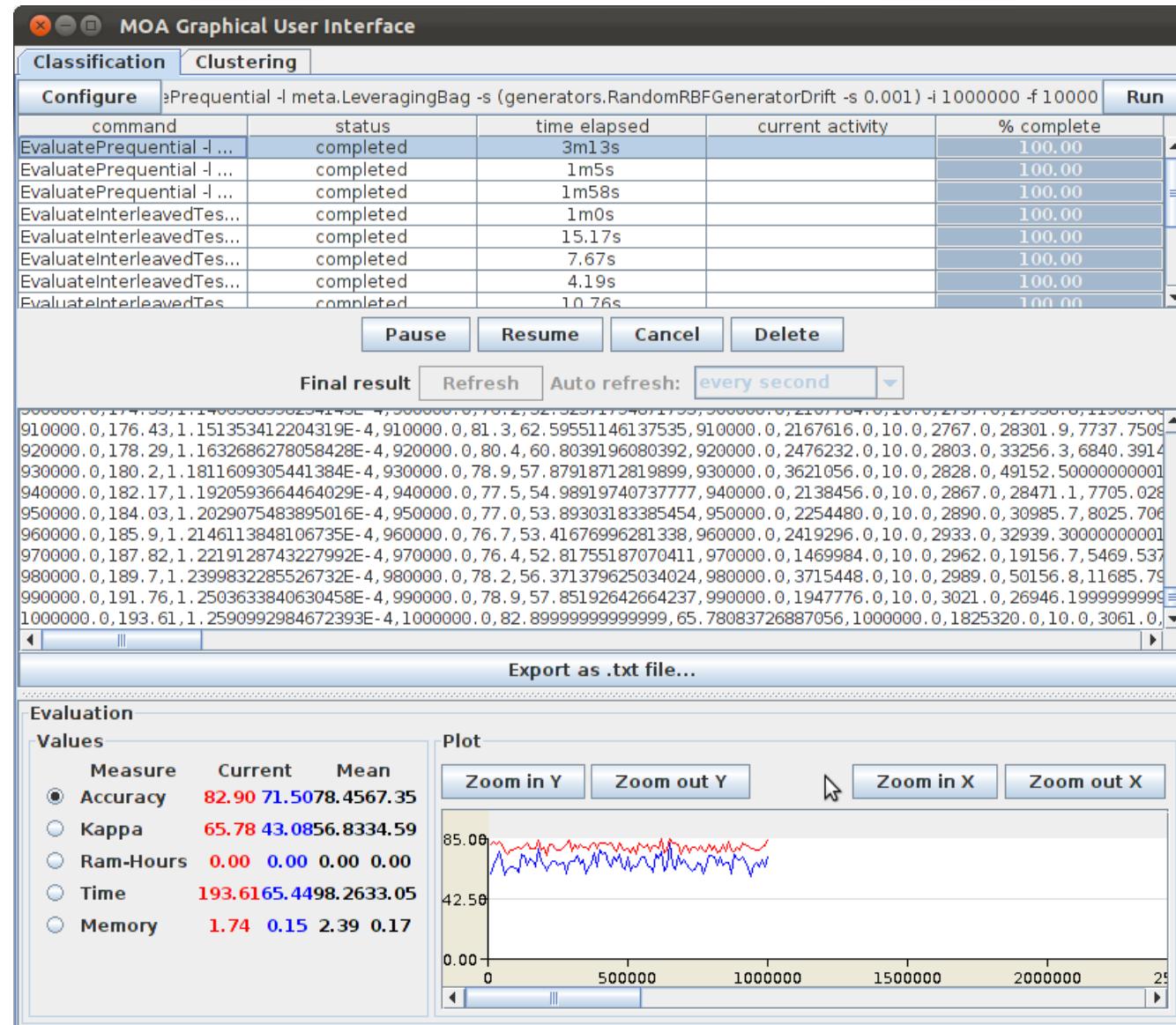
MOA – *Massive Online Analysis*

- MOA é um *framework* para mineração de fluxo de dados.
- Ele inclui ferramentas para avaliação e um conjunto de algoritmos de aprendizado de máquina
- Relacionado ao projeto WEKA , também é escrito em Java, escalável para problemas maiores.
- Pode ser estendido com novos algoritmos de mineração e novos geradores de fluxo ou medidas de avaliação.
- O objetivo é fornecer um software para a comunidade de DM para *Big Data*.



(not only a flightless bird, but also extinct!)

MOA – Massive Online Analysis



Explore e compreenda seus dados

Minere seus próprios dados (usuários, clientes e negócio) e torne-os em informações úteis para apoiar o Processo Decisório.

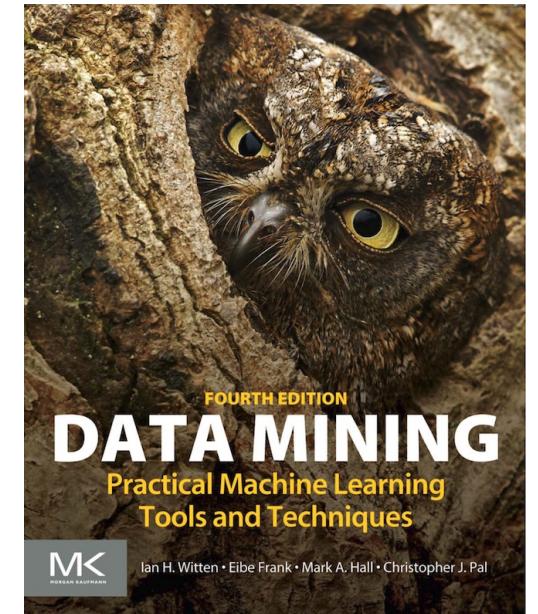
Comece a tomar decisões melhores utilizando as experiências passadas de maneira integrada com a sua aplicação.

Há vagas nesta área, basta ter o perfil adequado!



Onde conseguir mais informações?

- <http://weka.pentaho.com/>
- <http://www.cs.waikato.ac.nz/ml/weka/>
- Mineração de Dados - Conceitos, Aplicações e Experimentos com Weka
 - <http://www.lbd.dcc.ufmg.br/colecoes/erirjes/2004/004.pdf>
- Data Mining: Practical Machine Learning Tools and Techniques, Fourth Edition
- KDnuggets
 - news, software, jobs, courses,...
 - www.KDnuggets.com
- ACM SIGKDD – data mining association
 - www.acm.org/sigkdd



Contato

Obrigado a todos!

Prof. Marcos Vinicius Fidelis

fidelis@utfpr.edu.br

mvfidelis@uepg.br

