Pivotal

Petabyte Scale Data Warehousing Greenplum

MADlib -- In Database Parallel Analytics

Postgres Conf 2018

Marshall Presser
Craig Sylvester
Andreas Scherbaum
17 April 2018

Agenda

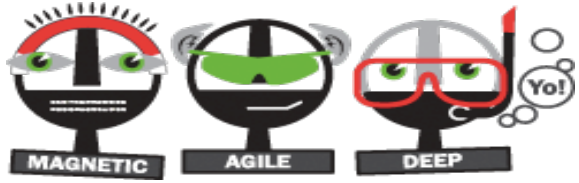
- Why MADlib® ?
- What is MADlib®
- Current MADlib® routines
- Small examples

Pivotal

Why MADlib® ?

- Most analytics is done by pulling the data to the analytic engine
  - Some enginges only accommodate in memory data
  - Sample data is the fall back – often not satisfactory
  - Time to pull often outweighs analytic processing time
- Developers sometimes write their own code
- There is a better way

Scalable, In-Database Machine Learning



Big Data Machine Learning in SQL for Data Scientists

Open Source, commercially usable BSD license	Supports Postgres, Pivotal Greenplum Database, and Pivotal HAWQ	Powerful analytics for Big Data

- Open Source Apache Top Level Progress
- Works on Greenplum and PostgreSQL
- In active development by Pivotal
- Downloads and Docs: http://madlib.apache.org/

Pivotal

MADlib Advantages

- Better parallelism
  - Algorithms designed to leverage MPP and Hadoop architecture
- Better scalability
  - Algorithms scale as your data set scales
- Better predictive accuracy
  - Can use all data, not a sample
- Open source
  - Available for customization and optimization by user if desired

Performing a linear regression on 10 million rows in seconds

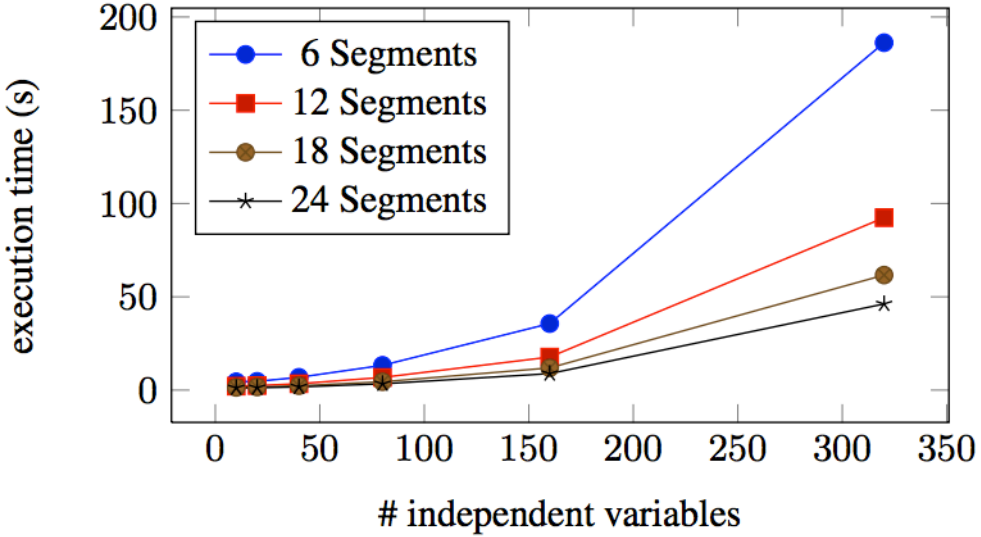
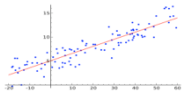
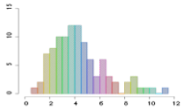
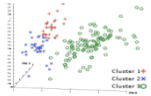


Figure 5: Linear regression execution times using MADlib v0.3 on Greenplum Database 4.2.0, 10 million rows

Hellerstein, Joseph M., et al. "The MADlib analytics library: or MAD skills, the SQL." Proceedings of the VLDB Endowment 5.12 (2012): 1700-1711.

Pivotal

MADlib Functions

Predictive Analytics Library

Supervised Learning

Regression Models

- Cox Proportional Hazards Regression
- Elastic Net Regularization
- Generalized Linear Models
- Linear Regression
- Logistic Regression
- Marginal Effects
- Multinomial Regression
- Ordinal Regression
- Robust Variance, Clustered Variance
- Support Vector Machines

Tree Methods

- Decision Tree
- Random Forest

Other Methods

- Conditional Random Field
- Naïve Bayes

Unsupervised Learning

- Association Rules (Apriori)
- Clustering (K-means)
- Topic Modeling (LDA)

Time Series

- ARIMA

Model Evaluation

- Cross Validation

Other Modules

- Conjugate Gradient
- Linear Solvers
- PMML Export
- Random Sampling
- Term Frequency for Text

Data Types and Transformations

- Array Operations
- Dimensionality Reduction (PCA)
- Encoding Categorical Variables
- Matrix Operations
- Matrix Factorization (SVD, Low Rank)
- Norms and Distance Functions
- Sparse Vectors

Statistics

Descriptive

- Cardinality Estimators
- Correlation
- Summary

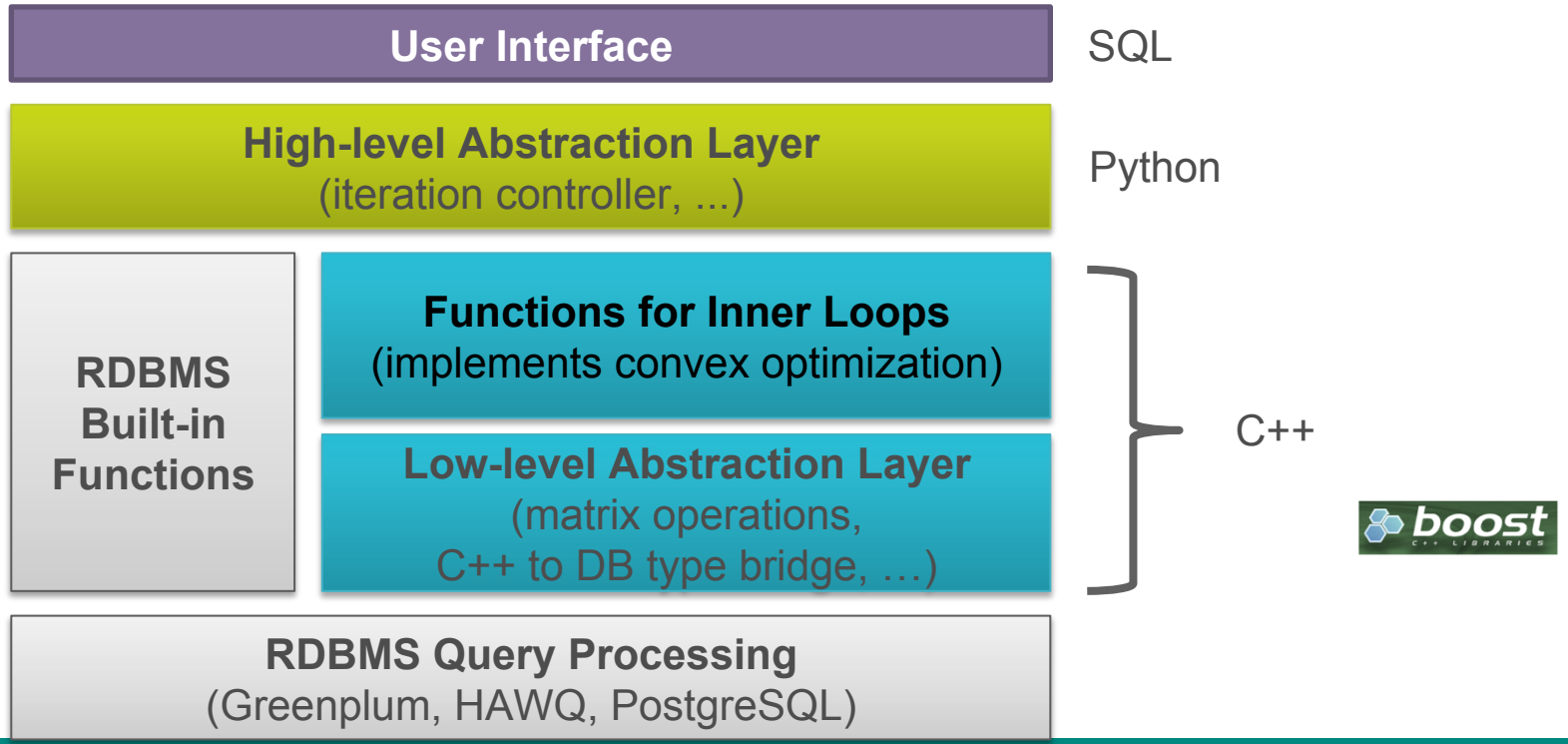
Inferential

- Hypothesis Tests

Other Statistics

- Probability Functions

MADlib Architecture



Pivotal

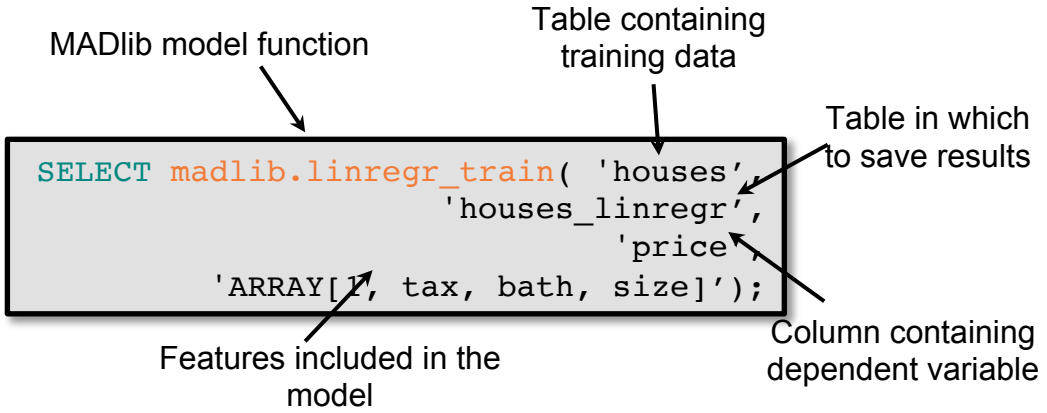
Pivotal Technology: Hadoop & HAWQ

- Performance through massive parallelism
- Automatic parallelization
  - Load and query like any database
  - Automatically distributed tables across nodes
- Analytics-oriented query optimization
- Scalable MPP architecture
  - All nodes can scan and process in parallel
  - Linear scalability by adding nodes

Pivotal

Calling MADlib Functions: Fast Training, Scoring

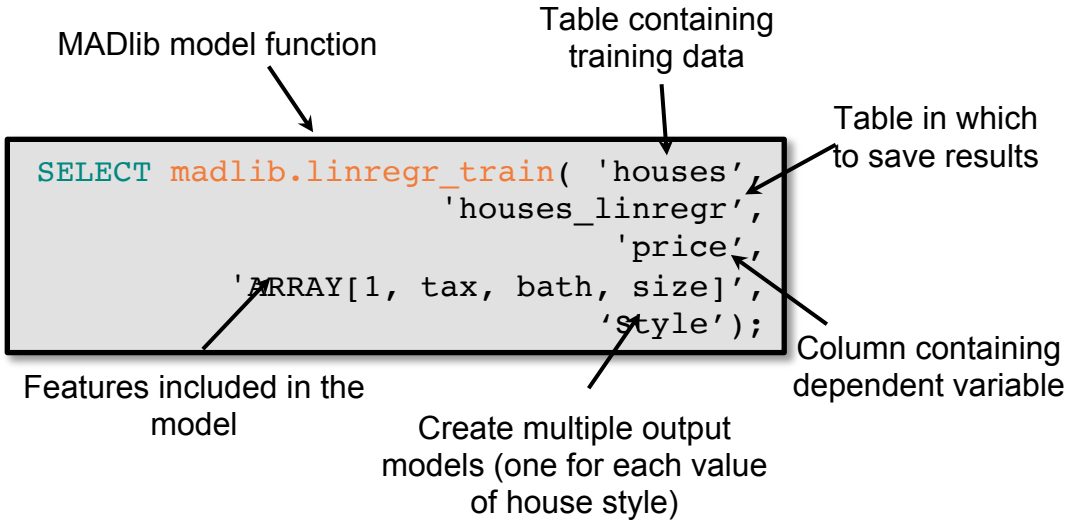
- MADlib allows users to easily and create models without moving data out of the systems
  - Model generation
  - Model validation
  - Scoring (evaluation of) new data
- All the data can be used in one model
- Built-in functionality to create of multiple smaller models (e.g. classification grouped by feature)
- Open-source lets you tweak and extend methods, or build your own



Pivotal

Calling MADlib Functions: Fast Training, Scoring

- MADlib allows users to easily and create models without moving data out of the systems
  - Model generation
  - Model validation
  - Scoring (evaluation of) new data
- All the data can be used in one model
- Built-in functionality to create of multiple smaller models (e.g. classification grouped by feature)
- Open-source lets you tweak and extend methods, or build your own



Pivotal

Calling MADlib Functions: Fast Training, Scoring

- MADlib allows users to easily and create models without moving data out of the systems
  - Model generation
  - Model validation
  - Scoring (evaluation of) new data
- All the data can be used in one model
- Built-in functionality to create of multiple smaller models (e.g. classification grouped by feature)
- Open-source lets you tweak and extend methods, or build your own

```
SELECT madlib.linregr_train( 'houses',
                             'houses_linregr',
                                      'price',
                    'ARRAY[1, tax, bath, size]');
```

MADlib model scoring function

```
                                    SELECT houses.*,
madlib.linregr_predict(ARRAY[1,tax,bath,size],
                                             m.coef
                                          )as predict
                        FROM houses, houses_linregr m;
```

Table with data to be scored

Table containing model

Pivotal

Pointer to Documentation

- General madlib documentation
  - http://madlib.apache.org/documentation.html
- Quick Start Guide
  - https://cwiki.apache.org/confluence/display/MADLIB/Quick+Start+Guide+for+Users

Pivotal