# Eclipse and the World of Data Science

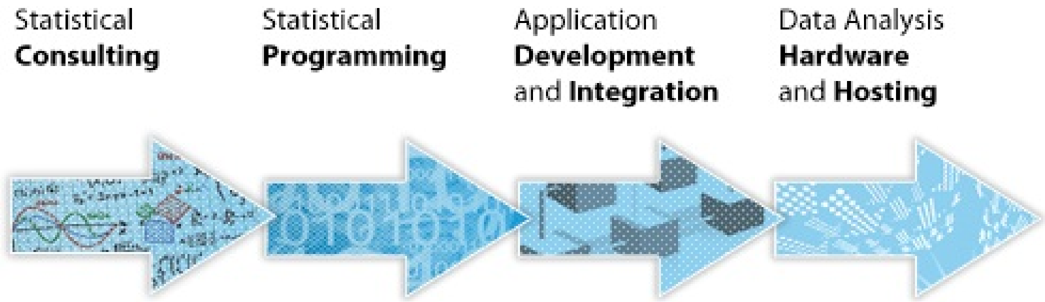
Tobias Verbeke (Open Analytics NV)

November 4, 2015

# Open Analytics

# Data Science Company

# Data Science Company



Statistical **Consulting**

Statistical **Programming**

Application **Development** and **Integration**

Data Analysis **Hardware** and **Hosting**

# Data Science

# What is a Data Scientist?

- "statistician who lives in Silicon Valley"
- "[…] a sexed up term for a statistician… Statistics is a branch of science. Data scientist is slightly redundant in some way and people shouldn't berate the term statistician" (Nate Silver)
- "someone who is better in statistics than any software engineer and better at software engineering than any statistician" (Josh Wils)

# What is a Data Scientist?

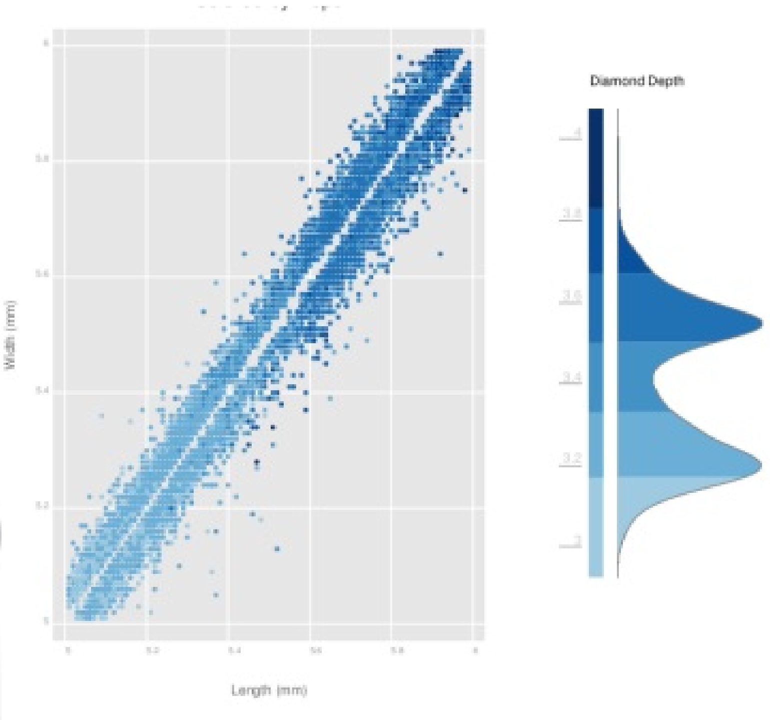
- "statistician who uses Eclipse" (Tobias Verbeke)
- we don't push buttons, we write code
- we use certain languages
- we need certain data structures and certain interfaces
- we produce certain output in certain ways

eclipse
READY

# Languages

# R

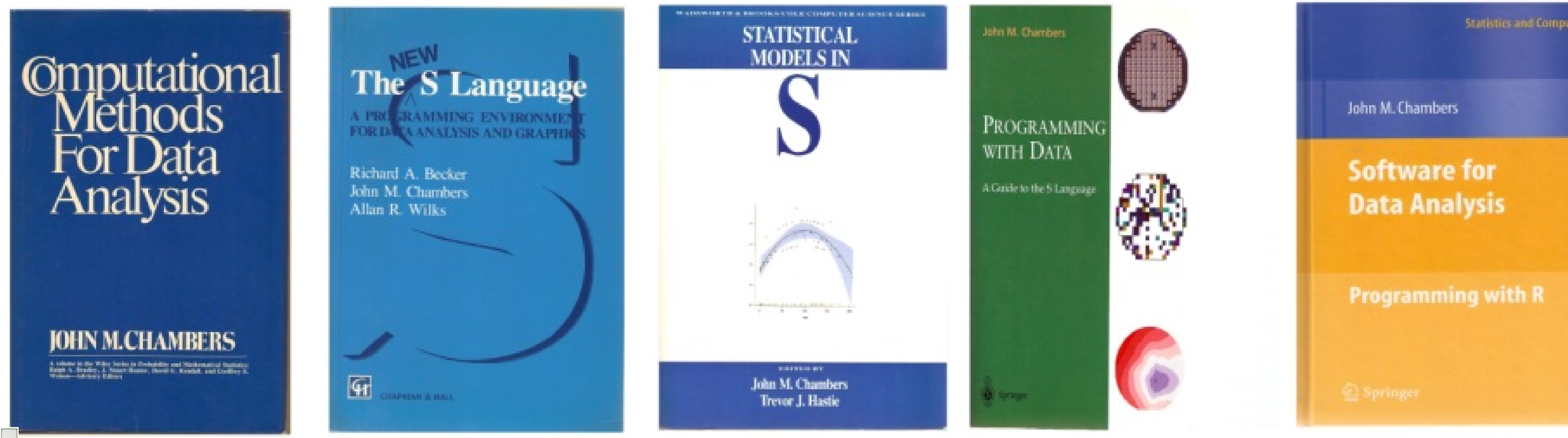
- environment for statistical computing and data analysis
- full-blown programming language, open source
- language designed with the modeler in mind
- model for a lot of the data science tools in other languages



# History of R

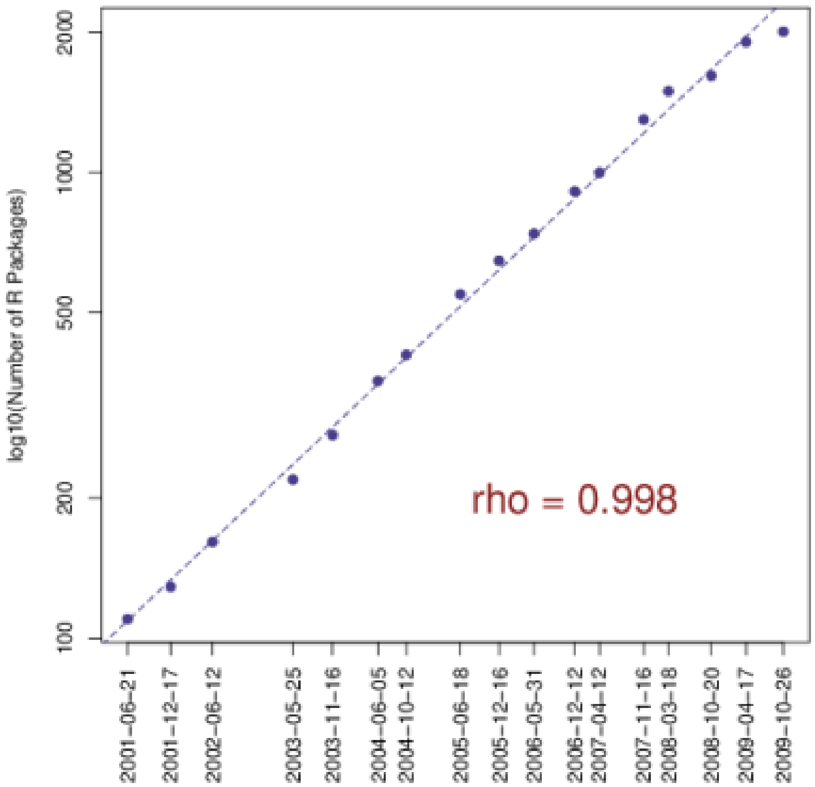
- S language at AT&T Bell Labs
- pioneering for interactive statistics (1975-1976)
- four landmark book publications (conceptual integrity)
- ACM Award 1998

"For the S system which has forever altered how people analyze, visualize and manipulate data"



# Who uses R?

- everyone (including Oracle, Microsoft, Google, HP, facebook, Pfizer, Bayer, Morgan Stanley, Ford, New York Times, John Deere, etc.)



# Data Structures

# data.frame

- not just arrays, but observations, labels, categorical data, ordinal data, numeric data
- built-in support for missing data (three-valued logic)
- neat indexing facilities

```
head(warpbreaks, n = 2)
```

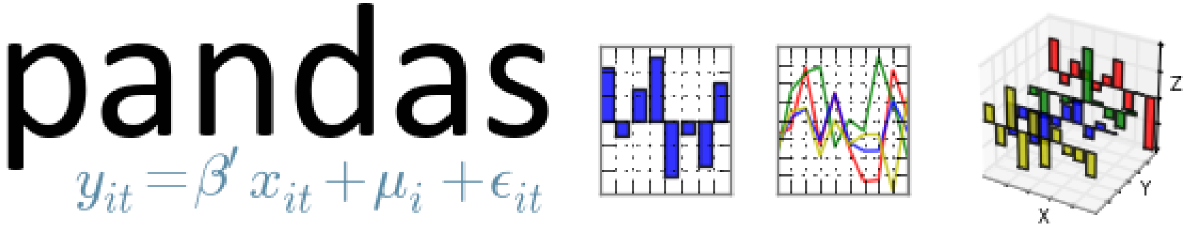
```
##   breaks wool tension
## 1     26    A       L
## 2     30    A       L
```

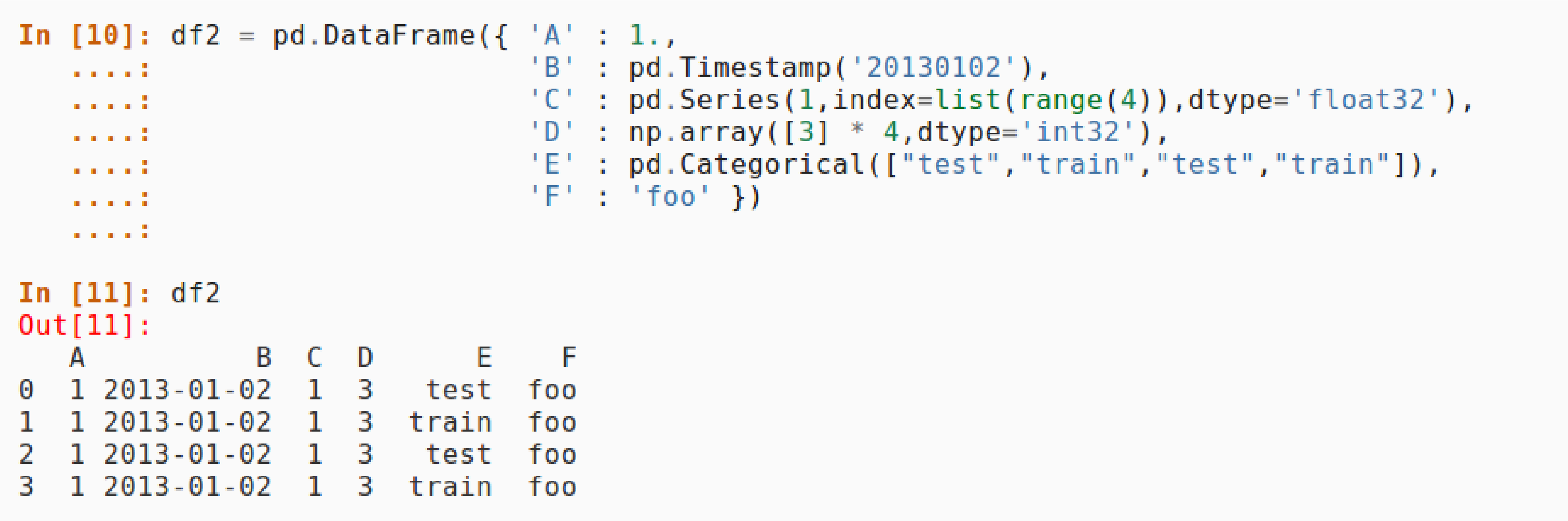
```
warpbreaks[warpbreaks$wool == "B" & warpbreaks$breaks < 15, 1:2]
```

```
##    breaks wool
## 29     14    B
## 50     13    B
```

# Python DataFrame

- pandas library for data manipulation and statistics
- defines a DataFrame object with integrated indexing





```
In [10]: df2 = pd.DataFrame({ 'A' : 1.,
   ....:                     'B' : pd.Timestamp('20130102'),
   ....:                     'C' : pd.Series(1,index=list(range(4)),dtype='float32'),
   ....:                     'D' : np.array([3] * 4,dtype='int32'),
   ....:                     'E' : pd.Categorical(["test","train","test","train"]),
   ....:                     'F' : 'foo' })
   ....:

In [11]: df2
Out[11]:
   A          B  C  D      E    F
0  1 2013-01-02  1  3   test  foo
1  1 2013-01-02  1  3  train  foo
2  1 2013-01-02  1  3   test  foo
3  1 2013-01-02  1  3  train  foo
```

# Spark DataFrame API

Quote from the 2015 Bossies:

The sweet spot for Spark continues to be machine learning. Highlights since last year include the replacement of the SchemaRDD with a Dataframes API, similar to those found in R and Pandas, making data access much simpler than with the raw RDD interface.

In the mean time, one can also use Spark interactively from an R terminal.

# DSL for modeling

# Turn Ideas into Software

- from mathematical idea to software

```
response ~ predictors
Fuel ~ Power + Weight
Fuel ~ Weight + sqrt(Power)
Fuel ~ poly(Weight, 3) + sqrt(Power)
Fuel ~ Power + sqrt(Weight) + Power:sqrt(Weight)
Fuel ~ Power * sqrt(Weight)
Fuel ~ Power * sqrt(Weight) + Type
Fuel ~ s(Power) + s(Weight)
```

- interfaces designed with the modeler in mind ('formula interface')

# Turn Ideas into Software (contd.)

```
lm(weight ~ group)
glm(lot1 ~ log(u), data = clotting, family = Gamma)
rpart(Kyphosis ~ Age + Number + Start, data = kyphosis)
gam(y ~ s(x0) + s(x1) + s(x2), family = poisson)
gee(breaks ~ tension, id = wool, data = warpbreaks, corstr = "AR-M", Mv = 1)
lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
```

# Python

- statsmodels library, depends on patsy library

patsy

latest

Search docs

Overview

Quickstart

How formulas work

Docs » patsy - Describing statistical models in Python

Edit on GitHub

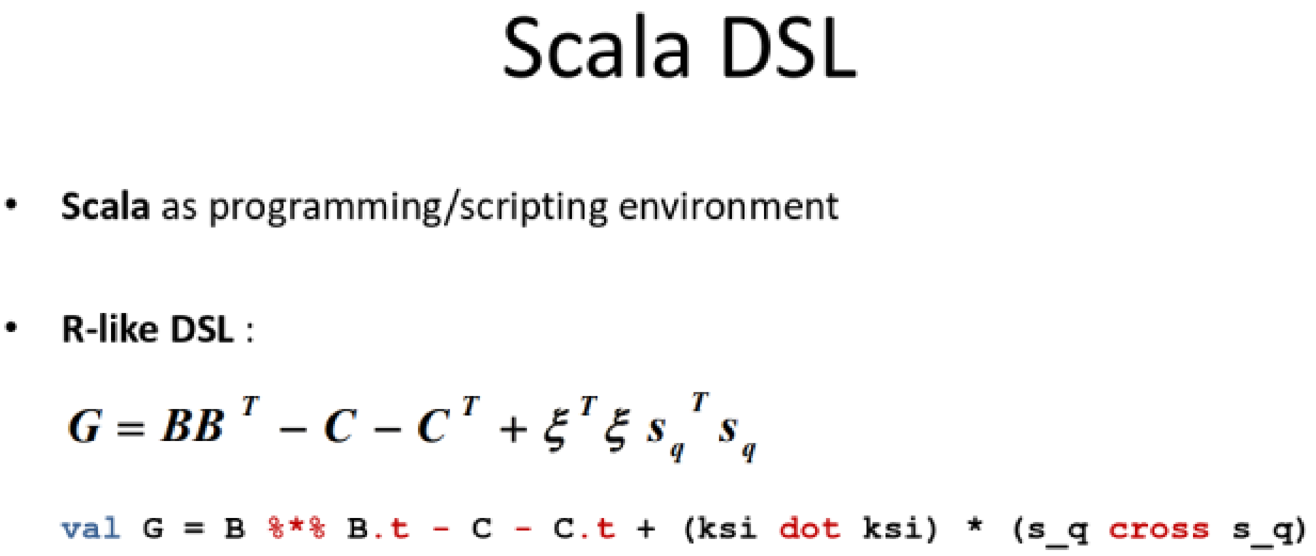
## patsy - Describing statistical models in Python

Contents:

```
ModelDesc.from_formula("Fuel ~ Power + Weight + Power:Weight")
```

# Apache Mahout DSL

- a little deeper than the formula interface
- distributed machine learning, moving away from MapReduce



## Scala DSL

- **Scala** as programming/scripting environment
- **R-like DSL** :

$$G = BB^T - C - C^T + \xi^T \xi \, s_q^T s_q$$

```
val G = B %*% B.t - C - C.t + (ksi dot ksi) * (s_q cross s_q)
```

(Courtesy of Sebastian Schelter)

# Demo

# Reproducible Research

# Reproducible research

- literate programming transposed to statistical practice
- analysis code and description of the analysis and results ("comments") in one single document
- push the button and the computer conducts the analysis, generates graphs and tables, includes these in the report and you're done

# Notebooks

- interactive form of a reproducible document
- code cells and non-code cells, interacts with R sessions etc.
- Jupyter notebook most succesful implementation

# Demo

# Science Working Group

# Building Blocks

- top-down: dawnsci, chemclipse, ICE
- bottom-up: triquetrum for scientific workflow engines, datasets, advanced visualization
- data science is the science of analyzing data independently of the scientific application domain
- room for more tooling that focuses on generic data science building blocks

SCIENCE
eclipse.org

# Some Examples

- Datasets project inspired on Numpy NDArray
- pandas, on top of Numpy, implements the data frames idea, could be the next step
- Scientific Reporting Mylyn docs extended to support Rmd documents, could be extended to pymd documents for reproducible reporting using Python

# IP in Science

- contributing back is in the researcher's DNA
- R is GPL, Python has a GPL-compatible license, a lot of LGPL out there etc.
- to build on the shoulders of giants, new ways need to be found to cohabit with these communities

# Conclusions

# Conclusions

- chances are you will see more and more data scientists
- by definition, they use Eclipse
- they will in all likelihood speak a mouthful of R
- time for woRld domination…

# Acknowledgements

- Stephan Wahlbrink (WalWare)
- Science WG Members

# Questions?

tobias.verbeke@openanalytics.eu

# Thanks!

eclipsecon Europe
Ludwigsburg, Germany, 3 - 5 November 2015
Evaluate the sessions at www.eclipsecon.org
+1
0
-1