Doctor n.0

PREDICTING HEART FAILURE USING ENSEMBLE LEARNING

Team n° 7
Stéphane Collot
Yannick Le Cacheux
Rishikesh Kulkarni

Objective

General objective:

- Given medical records of a patient over a given time period, predict whether he is likely to develop Heart Failure within a certain period of time

Methodology:

- Use training set as well as machine learning tools to train a classifier

Heart Failure

Why heart failure?

- Very common and deadly disease:
  - 5 to 10% of adults over 65 suffering from HF in developed countries (5 million in the US)
  - 35% risk of death during year after diagnostic, ~10% / year after
- Predictable
  - Some symptoms leading to HF can be detected early: diabetes, high blood pressure, cholesterol…
  - Early prevention can help decrease risk

Dataset

We are using ExactData set. Time period of 10 years from 2005 to 2015.

Some numbers about the dataset:

- 772.189 laboratory results, 313.921 diagnostics, 120.953 prescription medications, 269.301 vital signs

- 10.460 patients
  - including **4.712 patients diagnosed with Diseases Of The Circulatory System**

    ICD9 code within *[390.0; 495.0]*
  - including **879 patients diagnosed with *Congestive heart failure***

    ICD9 code *428.0*

Feature construction

Features taken into account: medications, diagnostics, lab results and Body Mass Index (BMI)

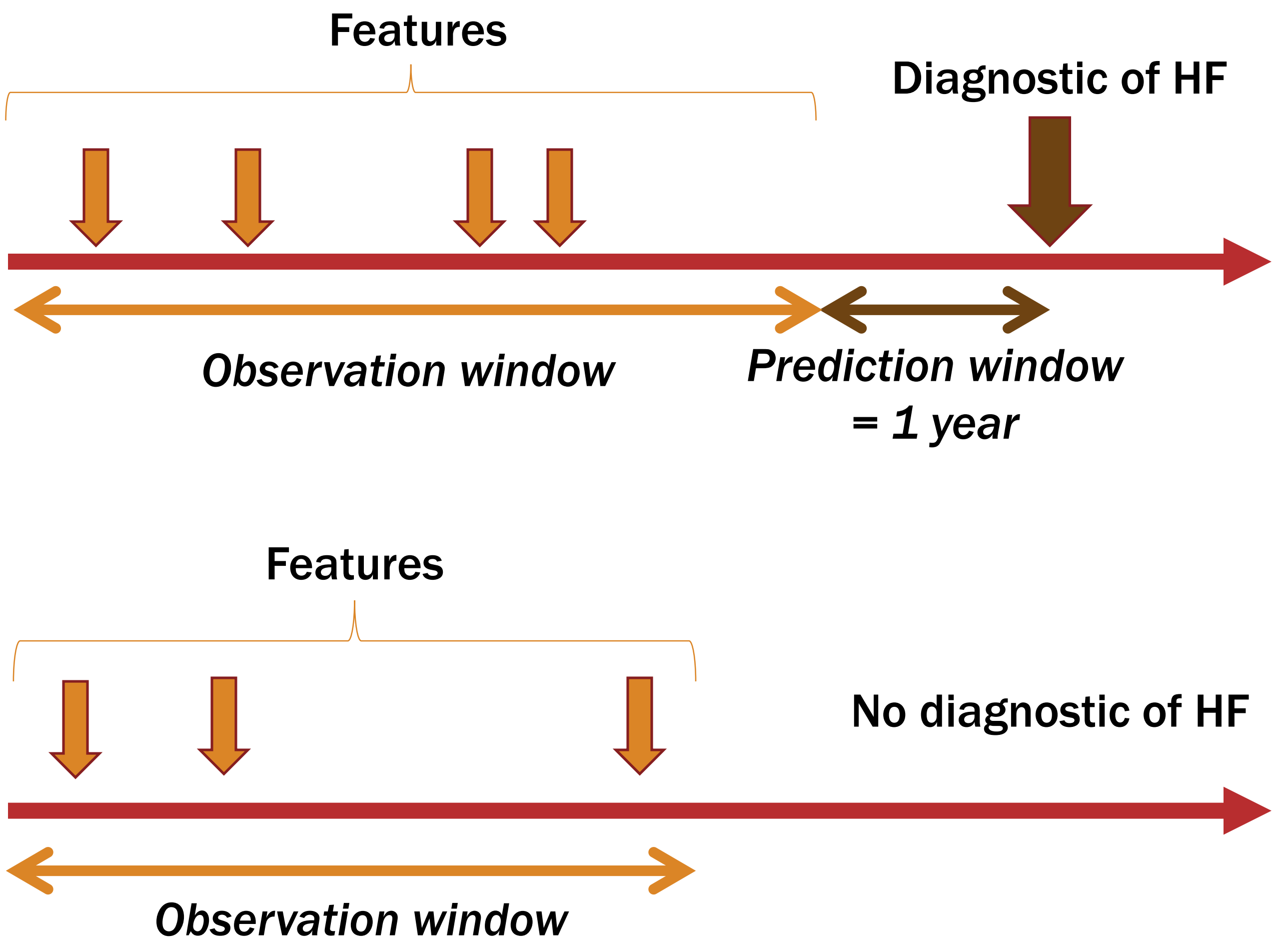
The number of features need to be the same for every patient

$$\begin{pmatrix} med1 \\ \vdots \\ diag1 \\ \vdots \\ lab1 \\ \vdots \\ BMI \end{pmatrix}$$

60 diff. medications + 52 diff. diagnostics + 100 diff. lab results + 1 BMI

= 213-dimensional features vector

Cohort construction



Labeled "YES"
(**presence** of HF diagnostic)

Labeled "NO"
(**absence** of HF diagnostic)

Training set construction

Only 5% of patients with HF in dataset

➢ 95% accuracy if the classifier always output “NO”

This is NOT what we are trying to achieve

➢Much more interesting to have a low false negative rate – helps prevent risk

Solution: Stratified Sampling

50% of patients with heart failure and 50% without in our training dataset

Classifiers

Binary classification: each patient is either a “YES” or a “NO”

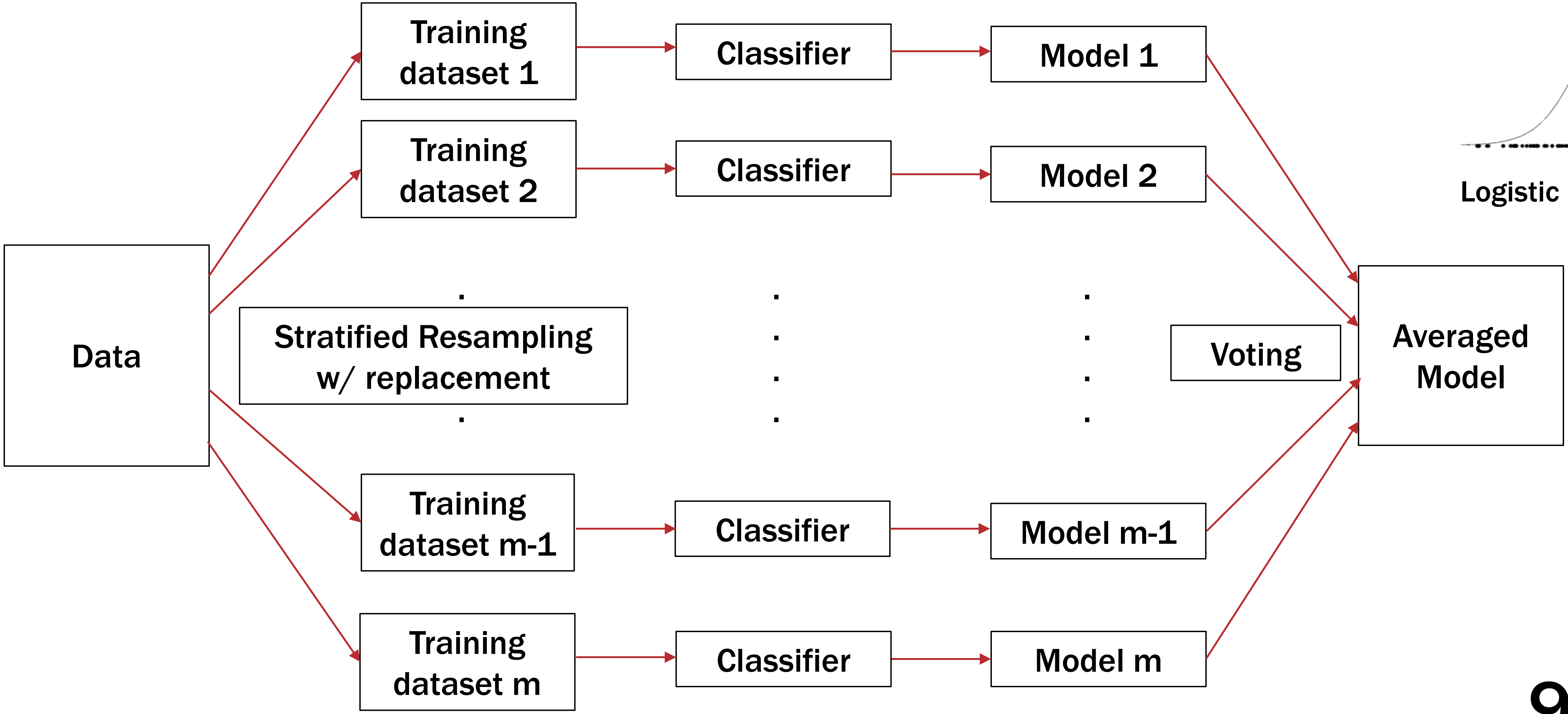
Base classifiers:

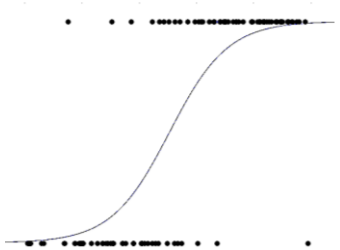
- Logistic regression
- Decision trees

➢ Unstable classifiers: use of **ensemble learning** to decrease variance and improve accuracy

  - Bagging for logistic regression
  - Random forest for decision trees

Logistic Regression - Bagging



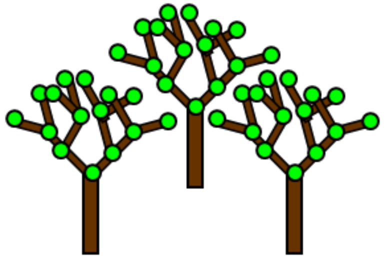


Logistic Regression

Decision Trees - Random Forest

Main idea

- Generation of a *large number of decision trees* introducing some *randomness* in the construction of the tree
- For each split in a tree, variables are chosen at random
- *Combination* of the results output by the different trees to calculate the final result
- Use of *vote* for final classification



Random Forest

Methods – Technology Stack

We are using

- Spark v1.3.0 (released Mar 13, 2015)
- Spark MLlib
- Spark ML (high-level API for machine learning pipelines)
- Scala
- Amazon Web Service
- GitHub

Cross Validation

- K-fold cross validation for validating performance
- K = 10 folds for cross validation
- Cross validation can also be used to determine optimum number of models in the bag
- B = 10, 20, 30
- Cross validation results used to pick out the classifier with best parameters overall

ML Pipeline

- ML Pipeline makes experimentation easy with Parameter Grid

Regularizer	Bag size	...
1.0	10	.
0.1	20	.
0.01	30	.
.	.	.

Strategy	Feature Subset	...
Classification	auto	.
Regression	sqrt	.
.	.	.

- Integrated two new algorithms with ML pipeline:
  - *class BaggedLogisticRegression*
  - *class RandomForestforPipeline*

Results – Confusion Matrices

- Logistic Regression - Bagging

96% accuracy

		Prediction	
		0 Healthy	1 Case Patient
Actual	0 Healthy	True negative 73	False positive 3
	1 Case Patient	**False negative** **3**	True positive 83

- Decision trees - Random Forest

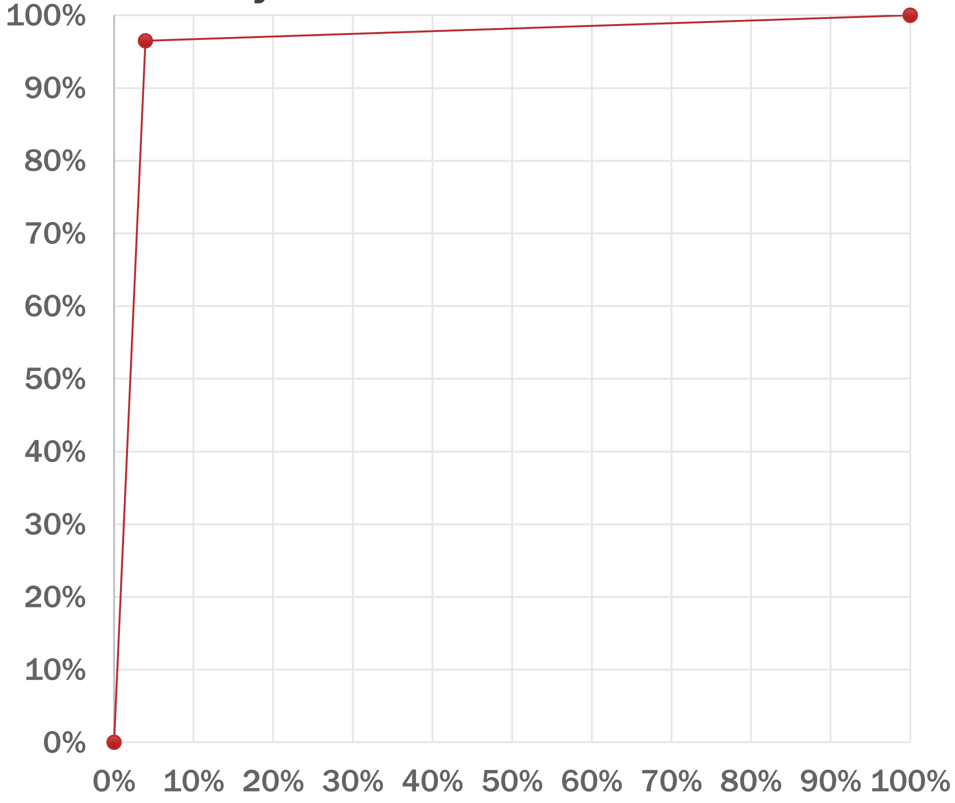
98% accuracy

		Prediction	
		0 Healthy	1 Case Patient
Actual	0 Healthy	True negative 74	False positive 2
	1 Case Patient	**False negative** **0**	True positive 86

Results – ROC curves

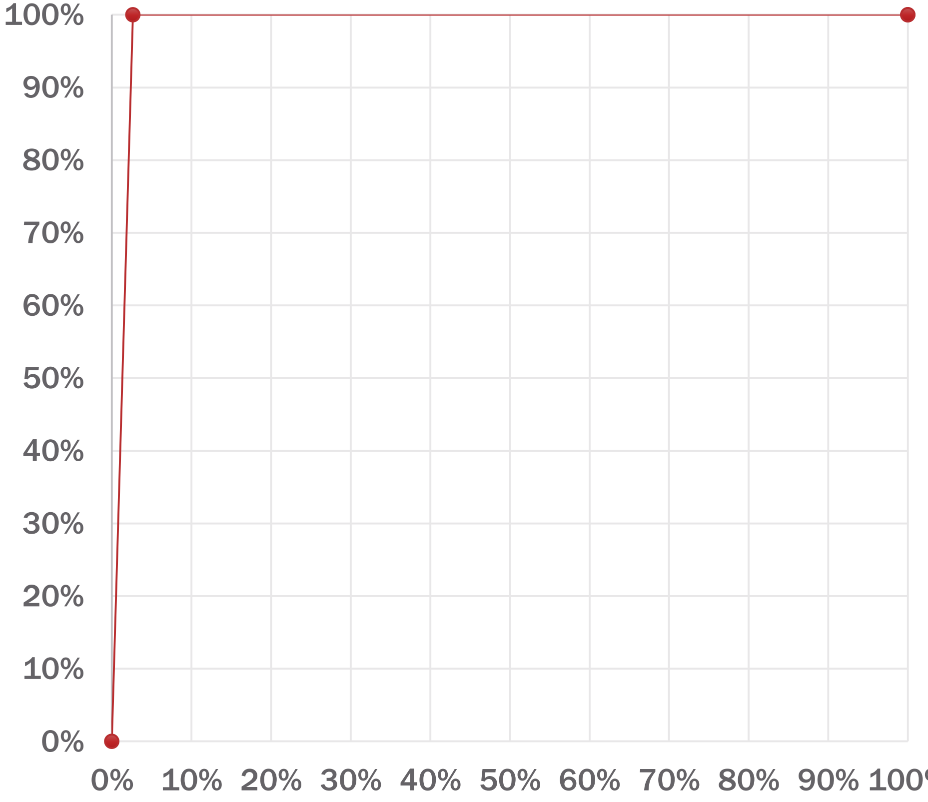
- Logistic Regression - Bagging

96% accuracy



- Decision trees - Random Forest

98% accuracy



Performance Evaluation

Amazon EC2

Performance evaluated on Amazon EC2 for different cluster parameters.

We tried:

- T2.micro 1 vCPU 1 GB of RAM
- M1.large 2 vCPU 7,5 GB of RAM
- M1.large 2 vCPU 7,5 GB of RAM x3 cluster

Bagging and random forest are parallelizable -> Scalable

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
my-spark-cluster3-master-i-42784c95	i-42784c95	m1.large	us-east-1d	running	2/2 checks
my-spark-cluster3-slave-i-b9784c6e	i-b9784c6e	m1.large	us-east-1d	running	2/2 checks
my-spark-cluster3-slave-i-5d784c8a	i-5d784c8a	m1.large	us-east-1d	running	2/2 checks

Challenges / Conclusion

Challenge:

- The data required scraping.

The results look too good:

- ExactData is not real data
- We don't have data before the observation windows
- Maybe there are medical features that are obviously correlated. We might need medical expertise to identify them.

References

1. Paradigm of Prediction: Predictive Analytics to Prevent Congestive Heart Failure
   Deborah Helen Selma, OJNI Volume 18, Number 2

2. Bagging Predictors
   Leo Breiman, Machine Learning, 24, 123–140 (1996)

3. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants
   Eric Bauer, Ron Kohavi, Machine Learning, July 1999, Volume 36, Issue 1-2, pp 105-139

4. Amazon EC2 instances
   http://aws.amazon.com/ec2/instance-types/

Doctor n.0

PREDICTING HEART FAILURE USING ENSEMBLE LEARNING

<u>Team n° 7</u>
Stéphane Collot
Yannick Le Cacheux
Rishikesh Kulkarni

Thank You
Any questions?