Noise and Outlier Detection

BORUT SLUBAN

DATA MINING AND KNOWLEDGE DISCOVERY

Anomalies?

- Errors in the data – noise
  - Animals of white color

- Exceptions or Outliers
  - Herd of sheep

Motivation

- **Noise** in data negatively affect data mining results.
  (Zhu et al., 2004)

- False medical diagnosis (**classification noise**) can have serious consequences
  (Gamberger et al. 2003)

- **Outlier** detection proved to be effective in detection of network intrusion and bank fraud.
  (Aggarwal and Yu, 2001)

Detecting noise and outliers

- Used for:
  - Improving machine learning performance through cleaning of training data
  - Data understanding and knowledge expansion by discovering potentially interesting exceptional cases in data

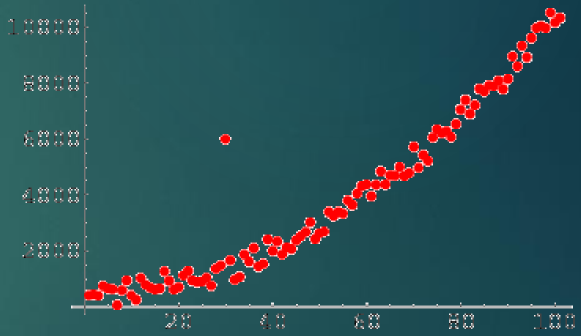
Detecting noise and outliers

- Nature
  - Follows certain patters
  - Adheres to the laws of physics
  - Is not random

Detecting noise and outliers

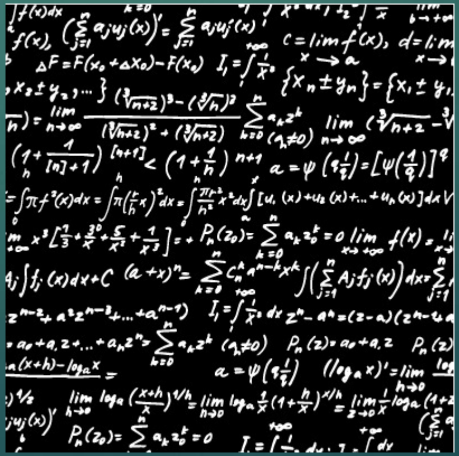
- Errors and exceptions are:

  - Inconsistencies with common patterns

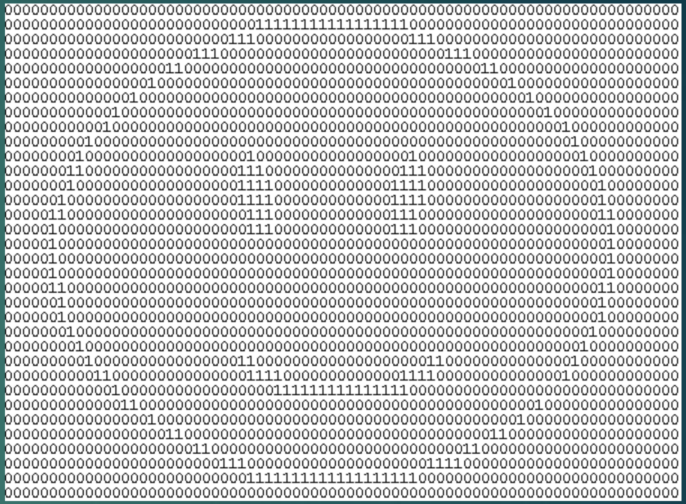


  - Great deviations from expected values

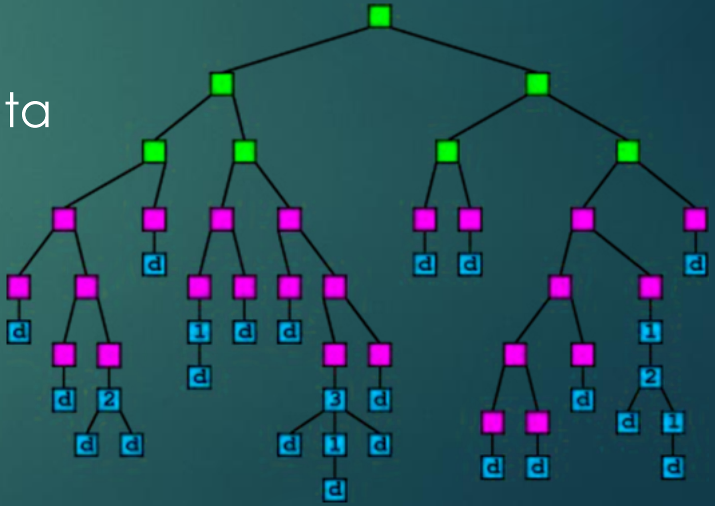
  - Hard to describe



Detecting noise and outliers



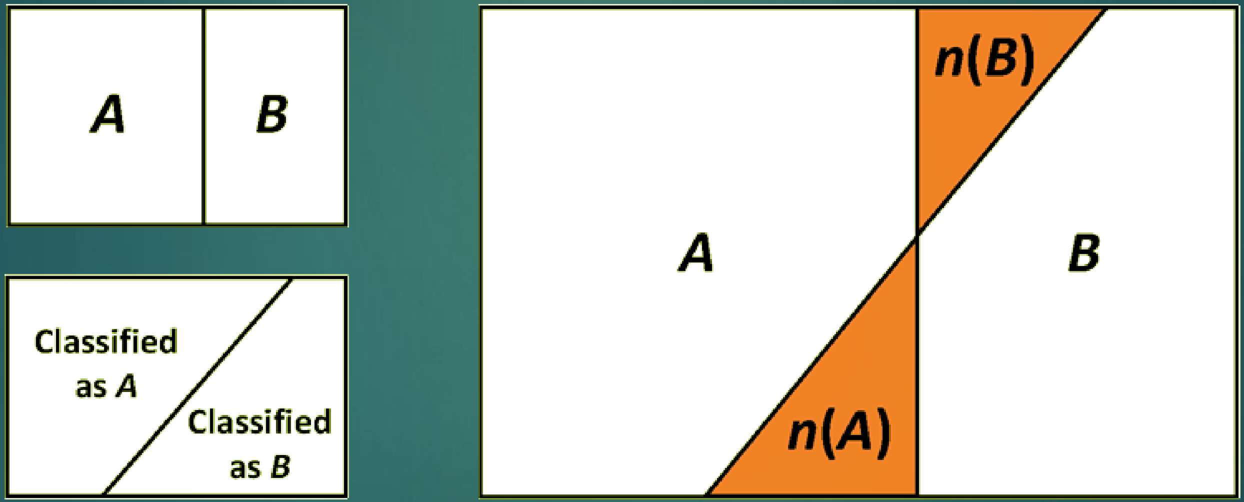
- Identify the “laws” of the data
- Build models
  - Patterns and rules = “laws” of the data



- **Errors and exceptions**
  - **Do NOT obey the laws (models)**

Classification noise filtering

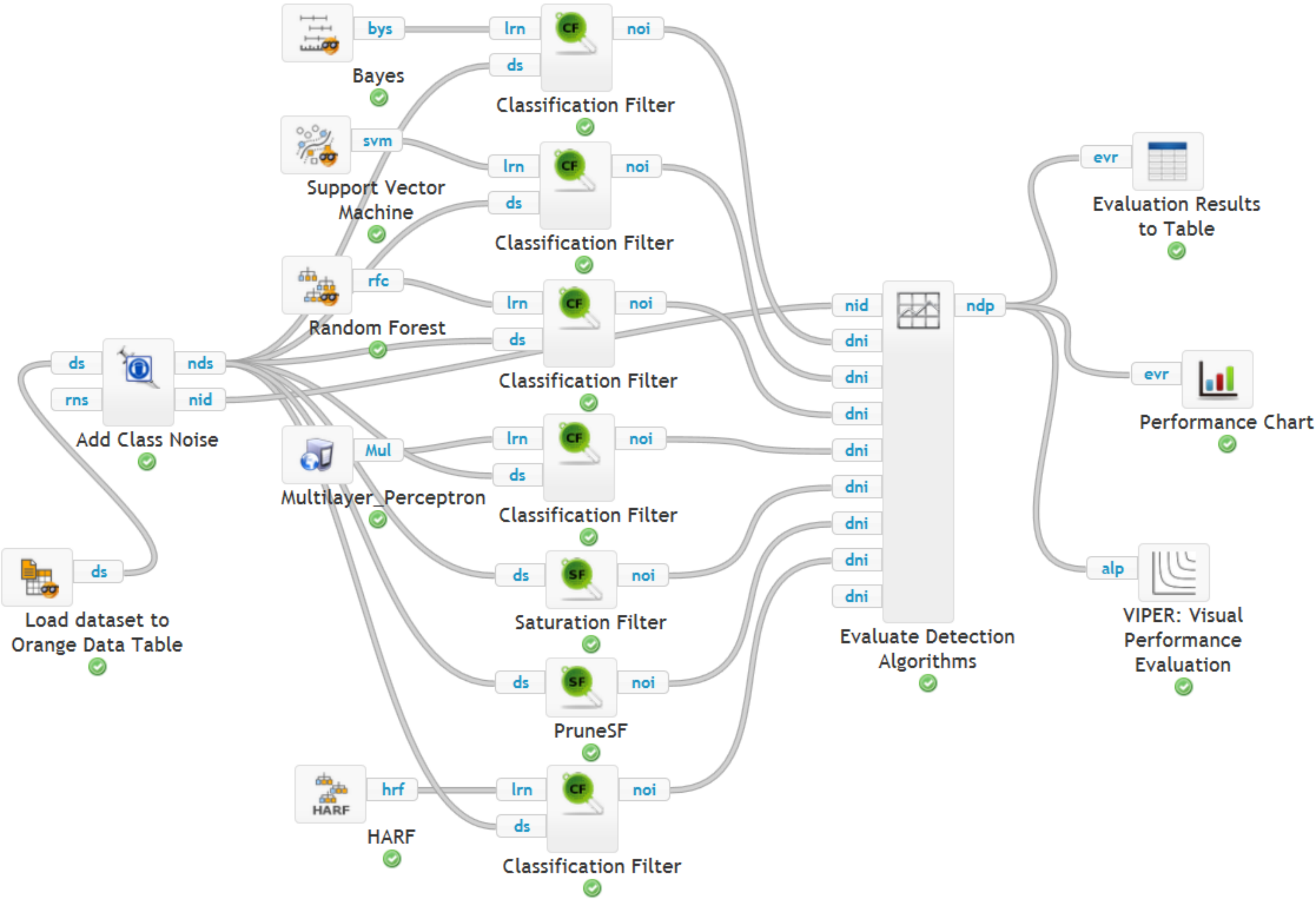
- Model the data
- What can't be modeled is considered noise



- Can use any learning algorithm

(Brodley & Friedl 1999)

Example Workflow

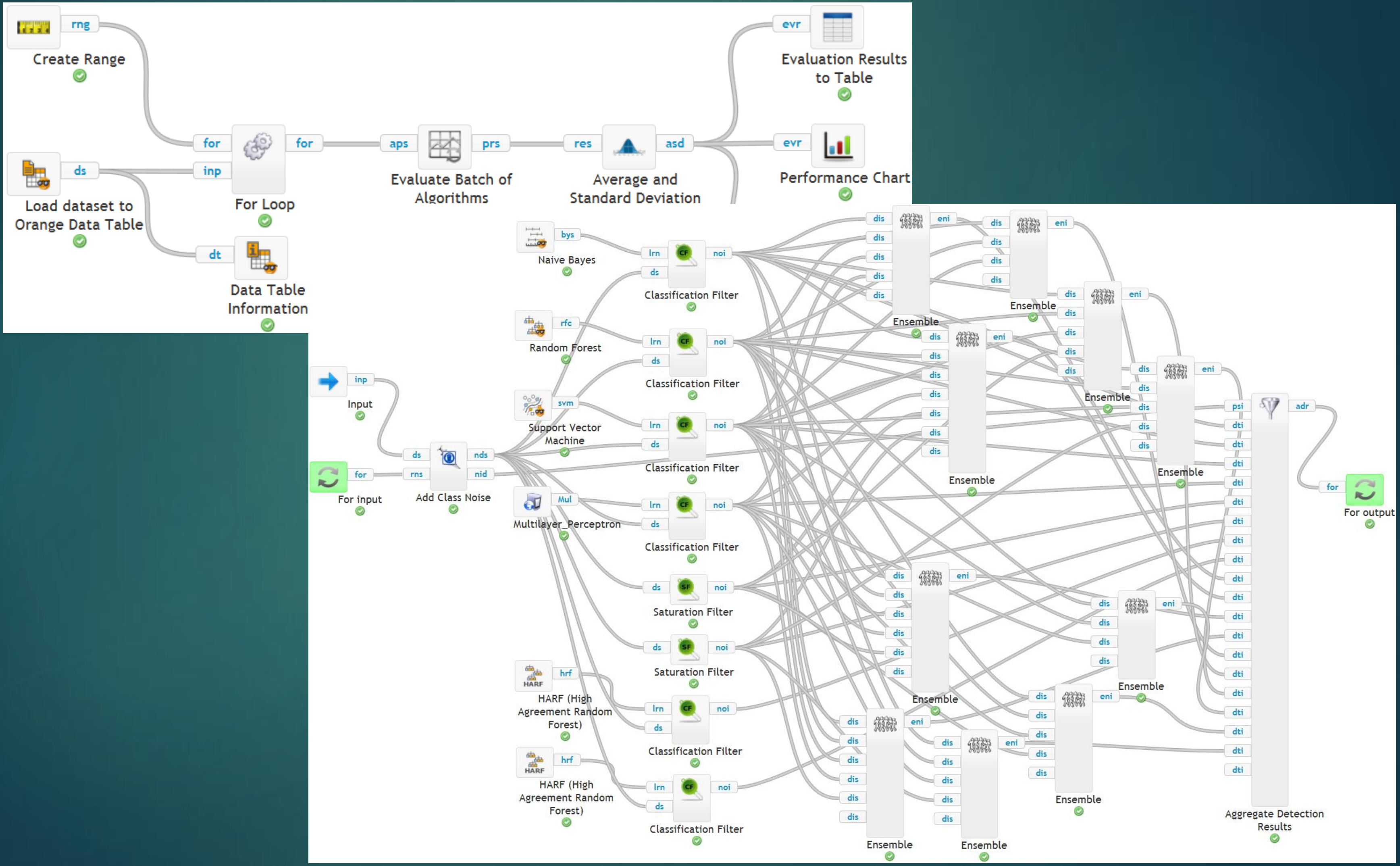


Ensembles

- Combine predictions of various models
- To overcome weaknesses or bias of individual models
- Averaging, Majority voting, Consensus voting, Ranking, etc.

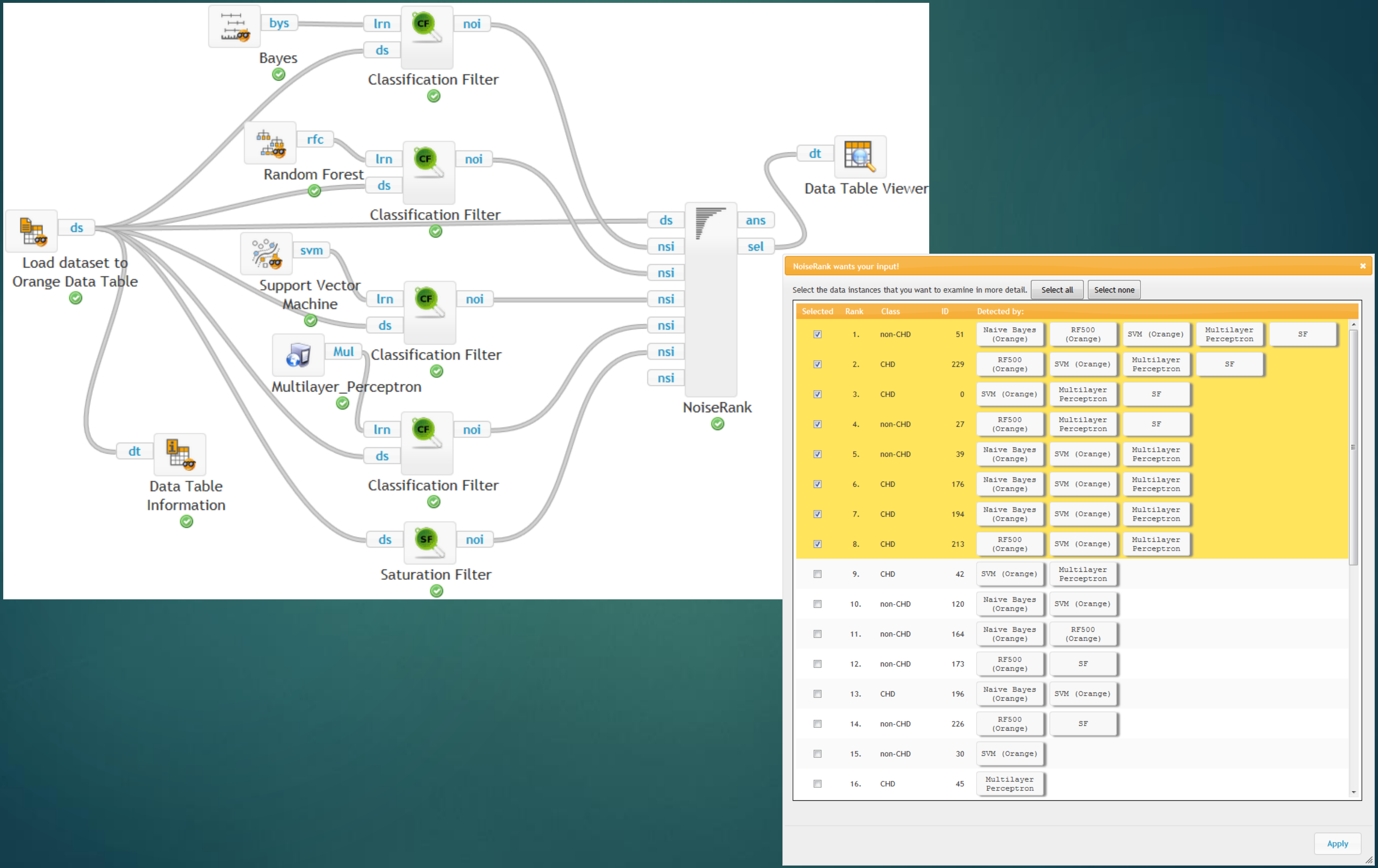
Example Workflows

- Ensembles of noise filters



Example Workflows

- NoiseRank



NoiseRank wants your input!

Select the data instances that you want to examine in more detail.

Selected	Rank	Class	ID	Detected by:				
☑	1.	non-CHD	51	Naive Bayes (Orange)	RF500 (Orange)	SVM (Orange)	Multilayer Perceptron	SF
☑	2.	CHD	229	RF500 (Orange)	SVM (Orange)	Multilayer Perceptron	SF	
☑	3.	CHD	0	SVM (Orange)	Multilayer Perceptron	SF		
☑	4.	non-CHD	27	RF500 (Orange)	Multilayer Perceptron	SF		
☑	5.	non-CHD	39	Naive Bayes (Orange)	SVM (Orange)	Multilayer Perceptron		
☑	6.	CHD	176	Naive Bayes (Orange)	SVM (Orange)	Multilayer Perceptron		
☑	7.	CHD	194	Naive Bayes (Orange)	SVM (Orange)	Multilayer Perceptron		
☑	8.	CHD	213	RF500 (Orange)	SVM (Orange)	Multilayer Perceptron		
□	9.	CHD	42	SVM (Orange)	Multilayer Perceptron			
□	10.	non-CHD	120	Naive Bayes (Orange)	SVM (Orange)			
□	11.	non-CHD	164	Naive Bayes (Orange)	RF500 (Orange)			
□	12.	non-CHD	173	RF500 (Orange)	SF			
□	13.	CHD	196	Naive Bayes (Orange)	SVM (Orange)			
□	14.	non-CHD	226	RF500 (Orange)	SF			
□	15.	non-CHD	30	SVM (Orange)				
□	16.	CHD	45	Multilayer Perceptron				

Try it out

- Noise filtering using ensembles (with performance evaluation)
  - http://clowdflows.org/workflow/245/

- NoiseRank
  - http://clowdflows.org/workflow/115/

- Clowdflows:
  - Noise Handling
  - Orange, Weka classification
  - Performance evaluation

- Need help or advice: borut.sluban@ijs.si