

# Metody eksploracji danych

## Laboratorium 2

Weka + Python + regresja

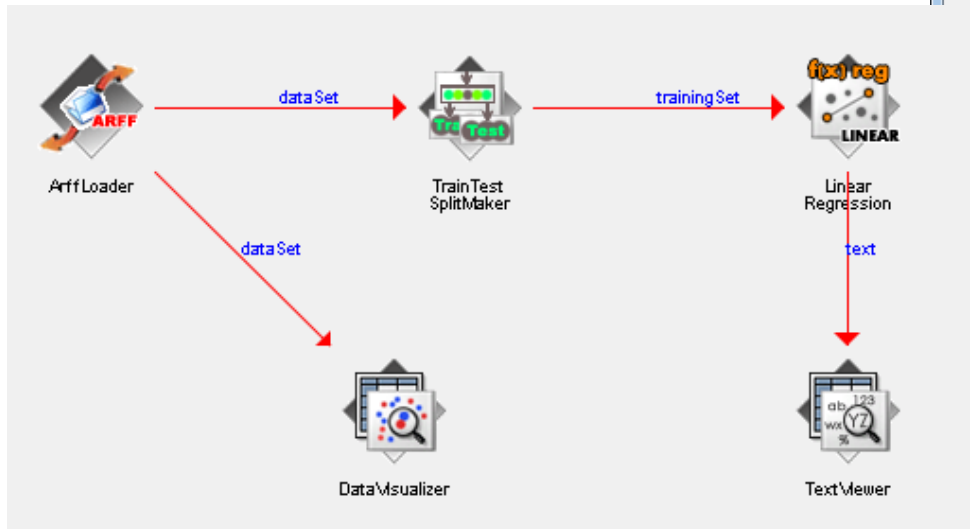
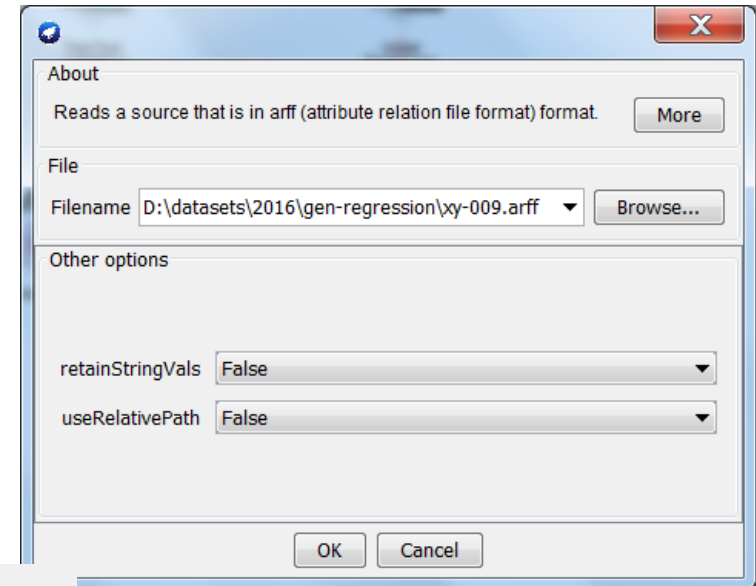
# KnowledgeFlow

- KnowledgeFlow pozwala na zdefiniowanie procesu przetwarzania danych
- Komponenty realizujące poszczególne czynności można konfigurować, np. podać nazwę pliku do odczytu, parametry operacji lub algorytmu
- Edytor zapewnia poprawność połączeń
- **Uruchom Weka korzystając z Java 1.7 Oracle (patrz wiki)**
- Uruchom z poziomu okna startowego

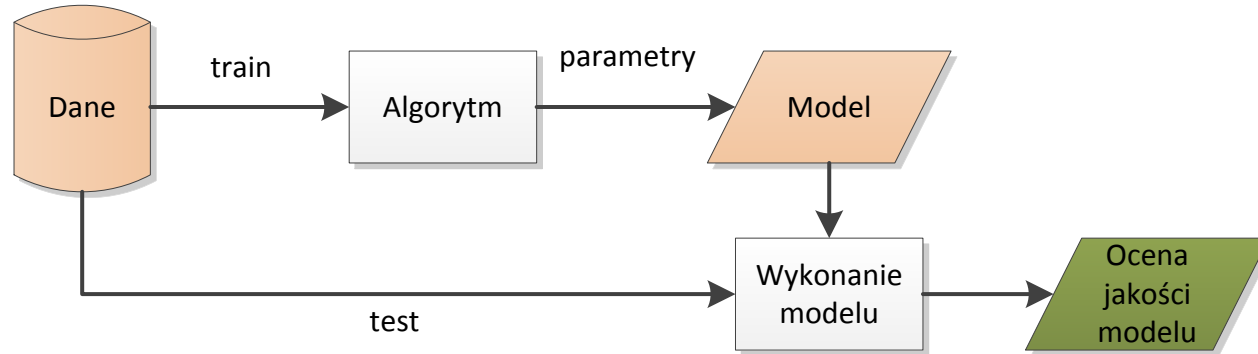


## 2.1 KnowledgeFlow – prosty proces

- Spróbuj zdefiniować pokazany proces
- Skonfiguruj ArffLoader tak, aby czytał zbiór danych **xy-009.arff**
- Kliknij prawym klawiszem na źródłowym komponencie, aby utworzyć połączenia
- Po uruchomieniu procesu ► możesz kliknąć na komponent i wybrać prawym klawiszem dostępne polecenia



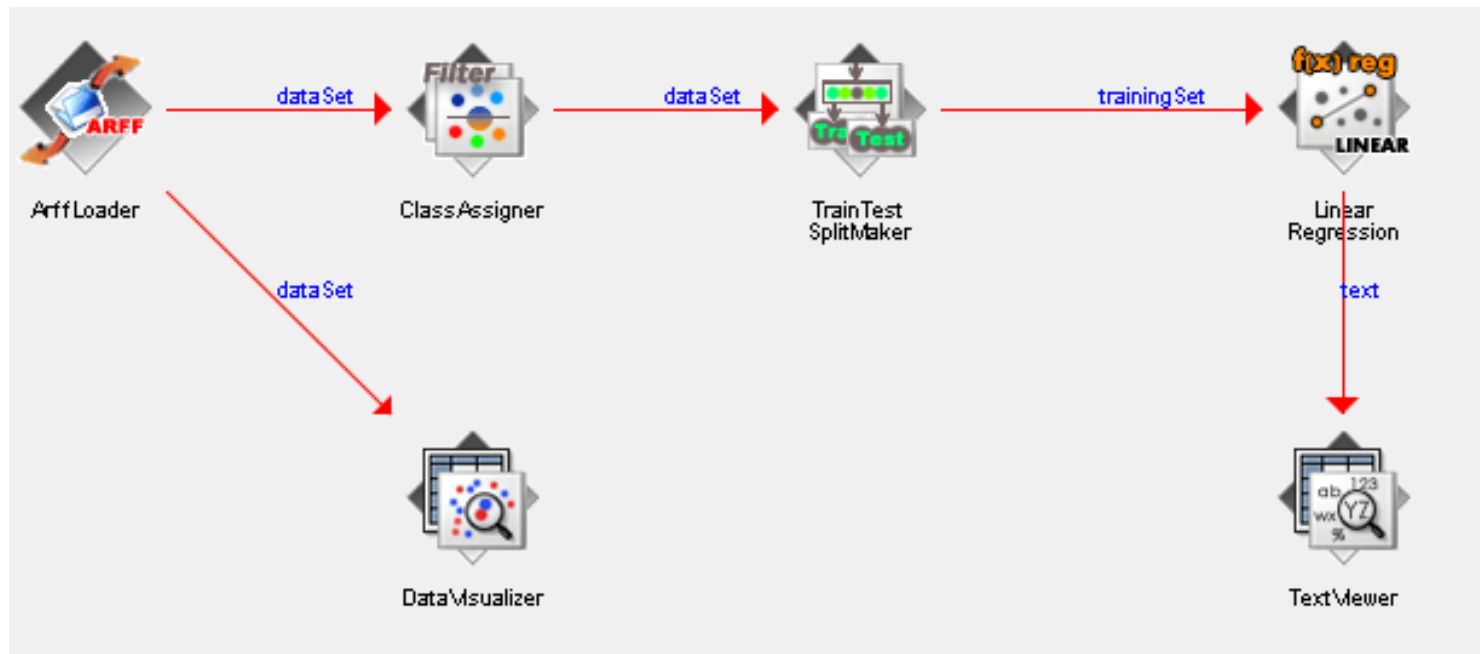
# KnowledgeFlow - po co TrainTestSplitMaker?



- Wyznaczanie modelu nie powinno ograniczać się wyłącznie do uczenia. Cały proces obejmuje także jego ocenę przeprowadzaną z użyciem innych danych niż użyte do uczenia.
- Edytor KnowledgeFlow nie pozwala na dostarczenie całego zbioru na wejście algorytmu uczenia (musimy dostarczyć trainingSet)

# KnowledgeFlow – poprawiony proces

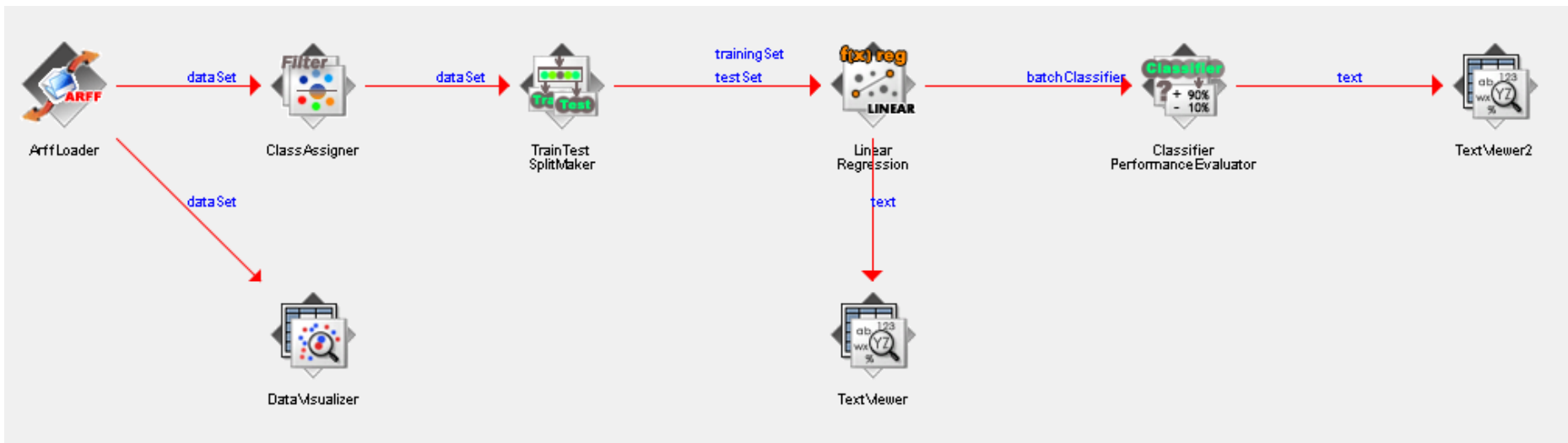
- Po uruchomieniu pojawiają się błędy? Algorytm nie wie, który atrybut jest wyjściowy.
- Zmień workflow



- ✓ Oczekiwany wynik:  
Linear Regression Model  
$$Y = -1.6615 * X + 13.2711$$

## 2.2 KnowledgeFlow – dodajemy ocenę

- Rozszerz workflow
- Nie zapomnij dołączyć przepływu testSet



# KnowledgeFlow – wyniki

Text

```
=== Evaluation result ===
```

```
Scheme: LinearRegression
```

```
Options: -S 0 -R 1.0E-8 -num-decimal-places 4
```

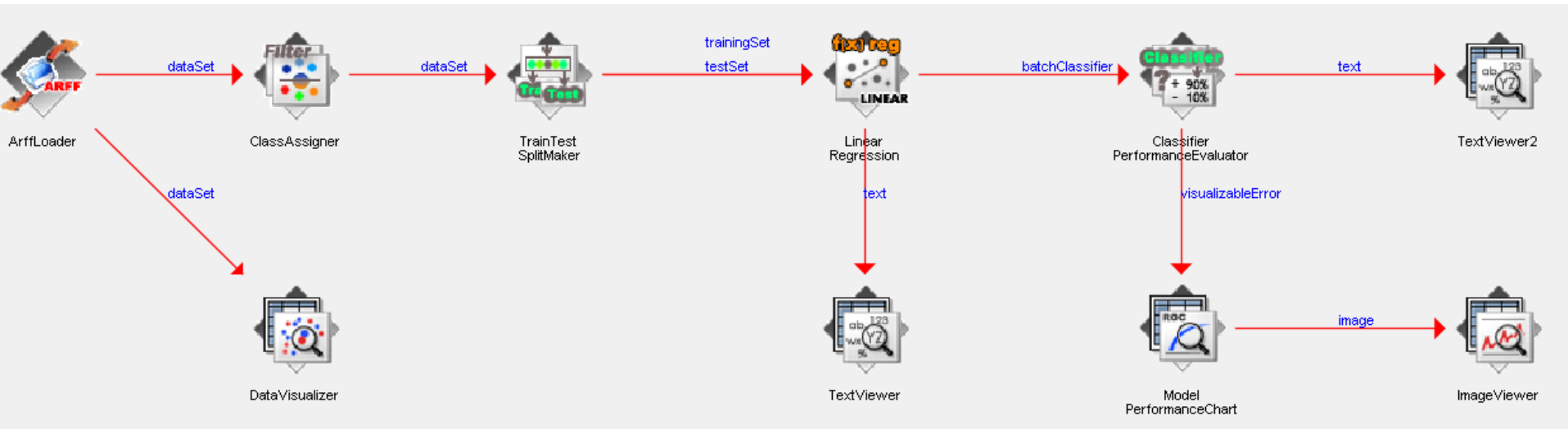
```
Relation: XY-(ellipse)-weka.filters.unsupervised.attribute.ClassAssigner-Clast
```

Correlation coefficient	0.9766
Mean absolute error	0.826
Root mean squared error	0.9689
Relative absolute error	21.5026 %
Root relative squared error	21.5537 %
Total Number of Instances	340

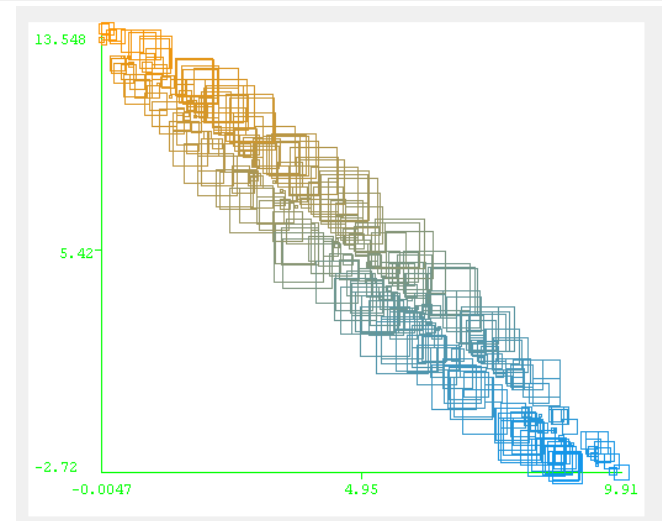
- Sprawdź, co oznaczają poszczególne wskaźniki:  
<http://stats.stackexchange.com/questions/131267/how-to-interpret-error-measures-in-weka-output>
- Dlaczego Total Number of Instances = 340 ?

# 2.3 KnowledgeFlow – dodatkowa wizualizacja

- Dodaj kolejne elementy i uruchom.



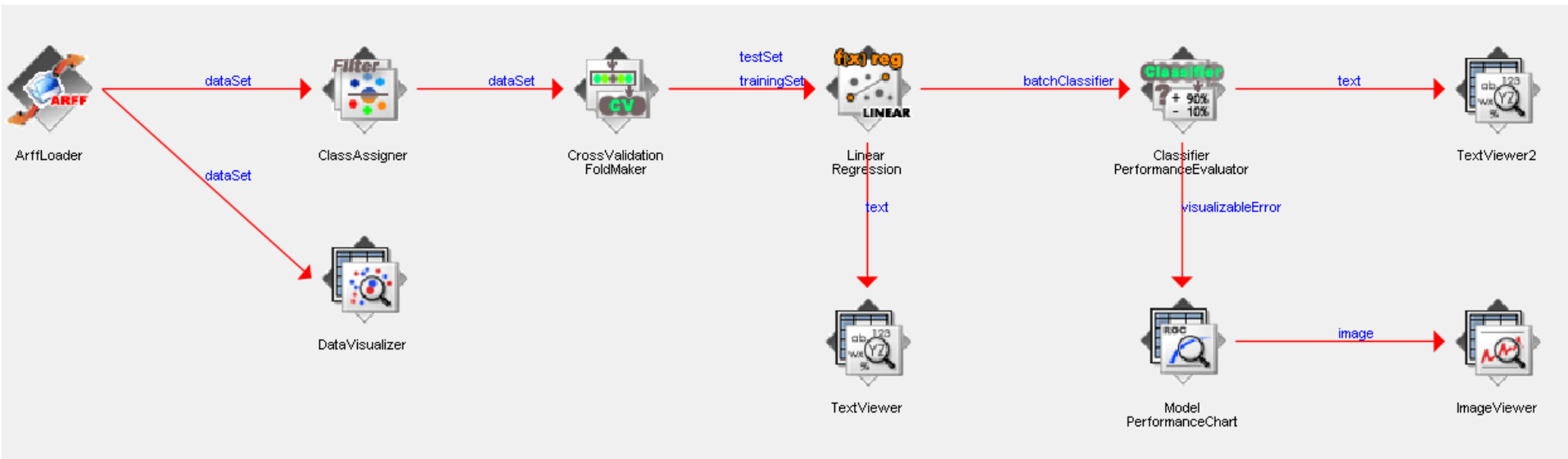
- Wyświetl wynik
- Dlaczego prostokąty oznaczające błędy układają się w elipsę?
- Skąd biały pasek pośrodku?





# 2.4 KnowledgeFlow – zmień sposób ewaluacji

- Użyj elementu CrossValidationFoldMaker



- CV (cross validation) polega na podziale zbioru na k (k=10) podzbiorów
- Uczenie i testowanie zachodzi w k iteracjach
  - k-1 części (folds) użyte do uczenia
  - 1 część użyta do testowania
- W sumie: do uczenia i walidacji użyte są wszystkie instancje
- Obliczane są średnie wyniki

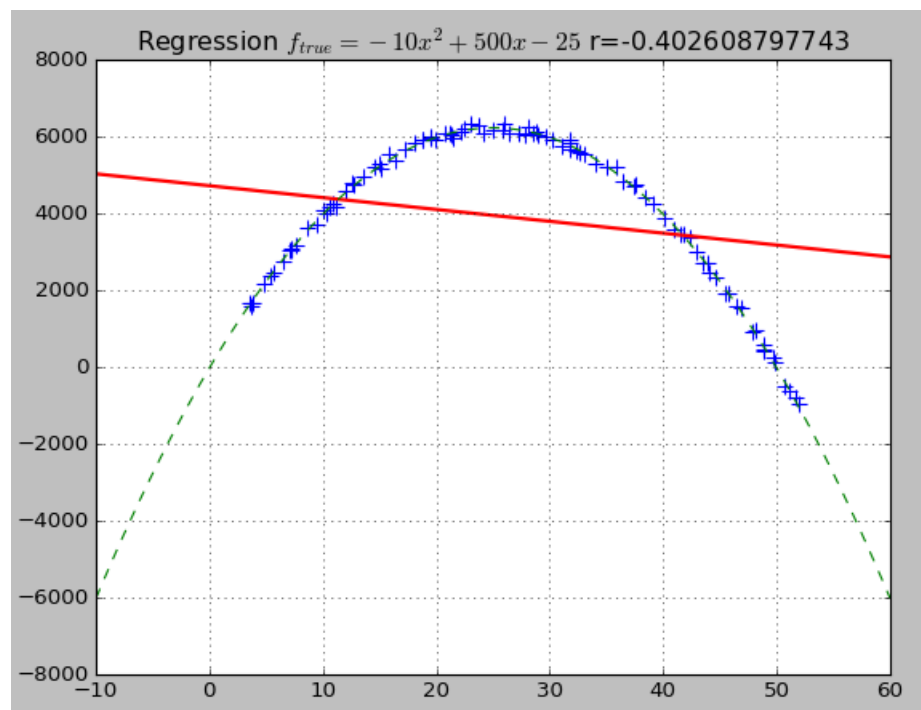
```
Text
=== Evaluation result ===

Scheme: LinearRegression
Options: -S 0 -R 1.0E-8 -num-decimal-places 4
Relation: XY-(ellipse)-weka.filters.unsupervised.attribute.ClassAssigner-Clas

Correlation coefficient          0.975
Mean absolute error             0.825
Root mean squared error         0.9704
Relative absolute error         22.1431 %
Root relative squared error     22.1937 %
Total Number of Instances      1000
```

## 2.5 Rozszerzanie zbioru cech

- Przetwarzany plik: **xy-004.arff**
- Uruchom zadanie regresji i odczytaj równanie krzywej
- Wyświetl dane + przebieg funkcji za pomocą odpowiednio zmodyfikowanego skryptu w języku Python
- Oczekiwany rezultat:

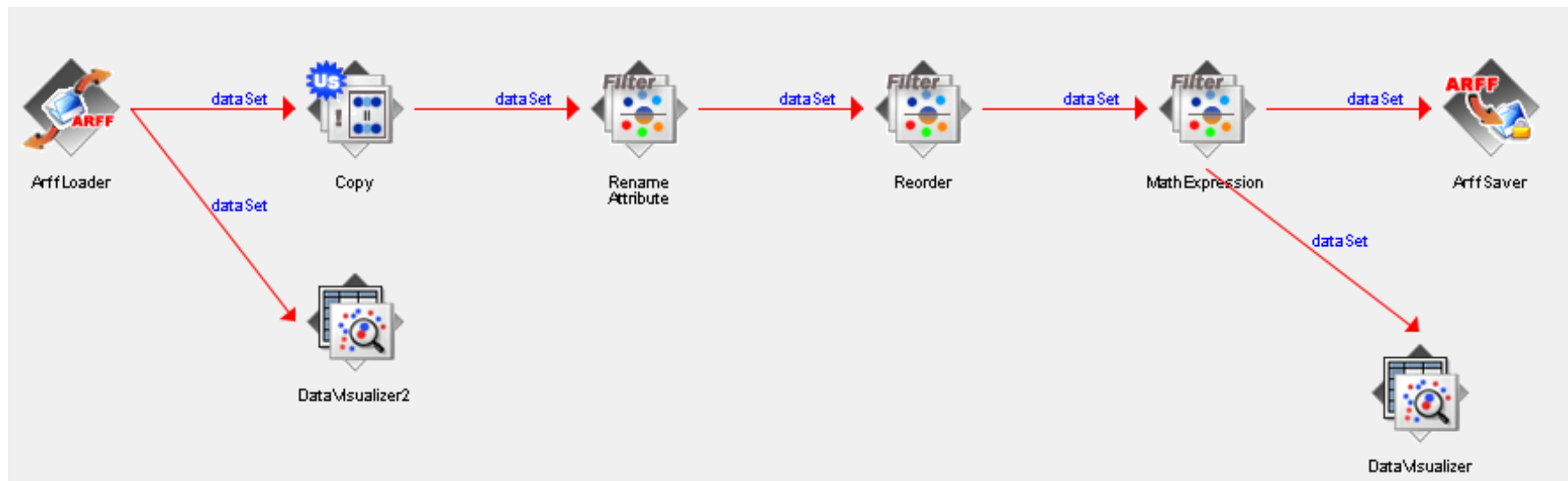


# Rozszerzanie zbioru cech

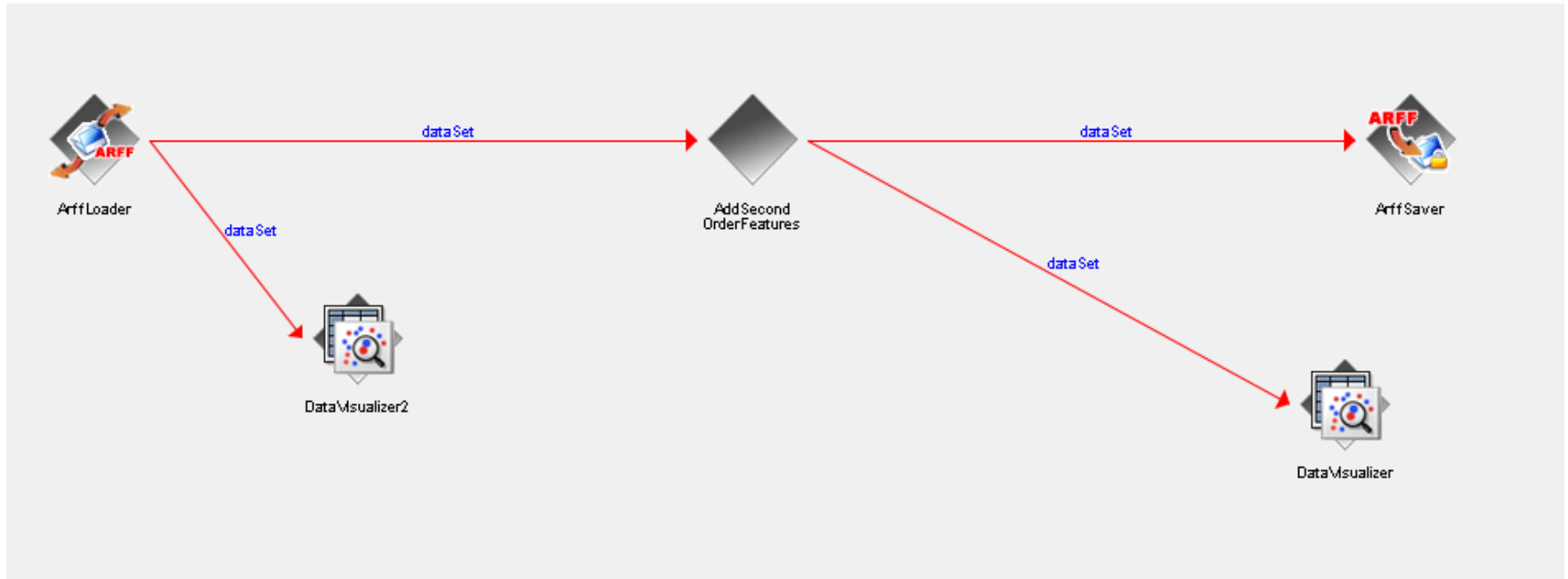
Zimplementuj poniższy proces konfigurując odpowiednio kolejne operacje:

- Skopiowanie atrybutu X
- Zmiana nazwy na X2
- Zmiana porządku tak, aby Y był ostatni
- Podniesienie X2 do kwadratu
- Zapis do pliku (aby sprawdzić poprawność).

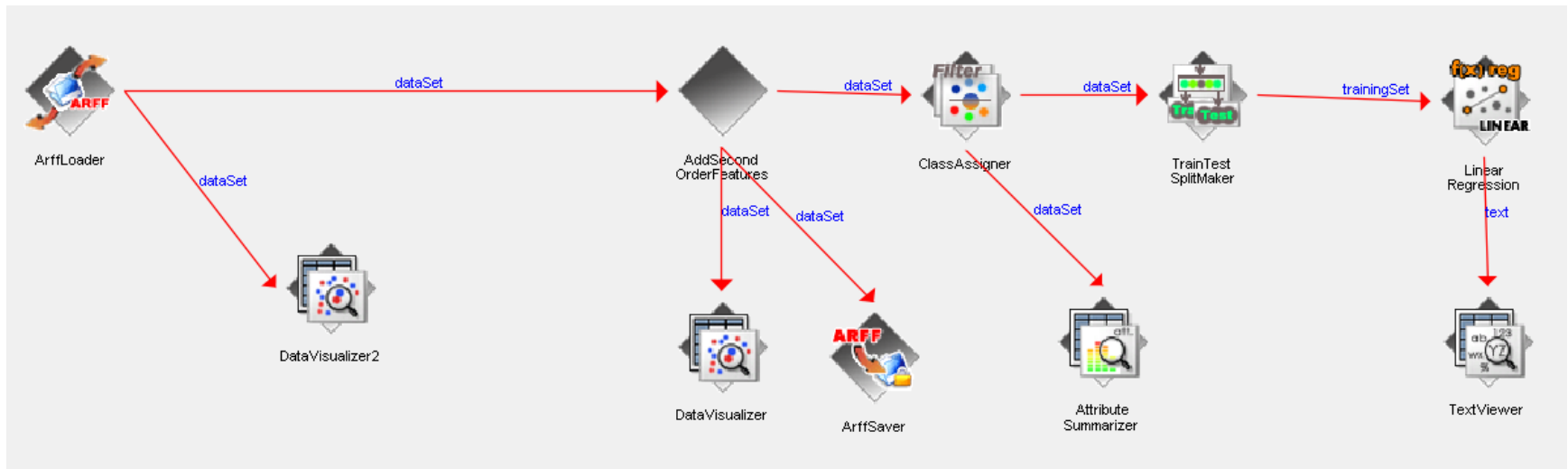
Wybierz np. nazwę **xy-004-01.arff**



# Zgrupuj fragment procesu



# Dodaj komponenty odpowiedzialne za przeprowadzenie regresji

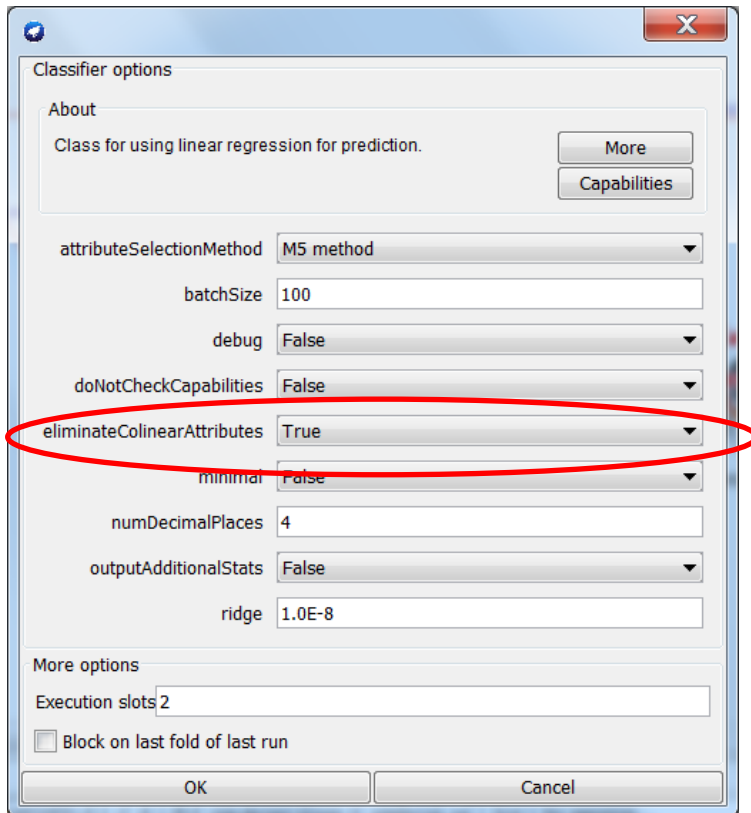


Uruchom i sprawdź wynik.

Jeśli ma on postać  $X^2 = a \cdot X \rightarrow$  Żle

Jeśli ma postać  $Y = \text{stała} \rightarrow$  Żle, ale to kwestia konfiguracji algorytmu

# Konfiguracja algorytmu regresji



- Zmień parametr na false
- Spójrz na wykres  $X/X_2$  W tym zakresie atrybuty są bliskie liniowym.
- ✓ Równanie regresji powinno mieć współczynniki bliskie rzeczywistej funkcji
- ✓ Sprawdź błędy walidacji. Błąd względny powinien być kilkuprocentowy
- ✓ Użyj kodu Pythona do narysowania danych oraz zależności  $x \rightarrow y$

## 2.6 Wprowadź cechy 3 stopnia

- Zbuduj KnowledgeFlow dla pliku **XY-005.arff**
- Wprowadź cechę 3 stopnia ( $x^3$ )
- Znajdź współczynniki regresji
- Wyświetl dane i wyznaczoną krzywą posługując się skrypcem w języku Python

## 2.7 Rozwiązanie w języku Python

- Przetwarzamy **xy-004.arff**: zapisz w KnowledgeFlow plik z dodatkowym atrybutem X2, np. jako xy-004a.arff
- Skopiuj zawartość pliku do skryptu Pythona lub wczytaj z pliku  
`x, x2, y = np.loadtxt(inp, delimiter=',', usecols=(0, 1, 2), unpack=True, skiprows=?)`
- Utwórz dwuwymiarową macierz cech:  
`features = np.stack((x,x2),axis=-1)`
- Dopasuj krzywą  
`regr = linear_model.LinearRegression()`  
`regr.fit(features, y)`
- Wyświetl wynik

