1. Načíst dataset
2. Zkontrolovat, zda nějaký sloupec nenese redundantní informaci
   1. Pokud ano, odstřelit ho
3. Zkontrolovat sloupce, zda mají všechna data
   1. Nemají?
      1. Je takových záznamů v poměru málo?
         1. Ano
            1. Odstranit
         2. Ne?
            1. Můžeme do sloupce vložit **průměrnou hodnotu** (Filter -> Unsupervized -> Atribute -> ReplaceMissingValues)

Pozor, nastaví všem sloupcům, hodí se tedy používat jako poslední operace.

1. Nastavíme sloupec, který chceme určit

# Algoritmy

* Baseline je zeroR
* Lazy.IBK – k nejbližších sousedů, počet sousedů se nastavuje v KNN
* **Stromové**
  + J48
  + RepTree
  + RandomForest
* Support vector machine
  + Functions - SMO

# Experimenter

Pro porovnání vícero algoritmů je potřeba přepnou testing with na paired T-Tester

# Úpravy sloupců

* **Číselná hodnota nabývá dvou stavů?**
  + NumerToBinary – Unsupervised – Attribute – NumbericToBinary
    - Nastaví se sloupec, kterého se to týká
* Podívat se na sloupce (nominal typu), zda jeden nepoužíváme F a Female pro označení ženy.
* **Odstranění záznamů s prázdnou hodnotou ve sloupci**
  + **Filter – unsupervised – instance (asi že se jedná o celý řádek) – RemoveWithValues**
    - Zadává se který atribut se má sledovat, zaškrtne se, jeslti se mají hledat prázdné sloupce, a nominalIndices = ‘’
* **Prázdná hodnota nahrazena střední hodnotou**
  + **Filter – unsupervised – attribute – replaceMissingValues**
    - Pozor, nahradí chybějící ve všech atributech, neváže se tedy konkrétně k jednomu sloupci
* **Diskretizace numerických hodnot**
  + Rozsahy převede na „kategorie“
  + **Filter – unsupervised – attribute – discretize**
    - **atributeIndices –** indexi atributů
    - **bins –** počet kategorií
    - **findNumBins = true** – sám zkusí odhadnou správný počet
* **Chybející hodnoty nahradit konstantou**
  + **Unsupervised – Attribute - ReplaceMissingWithUserConstant**
  + Konstanta se zadává do nominalStringReplaceMentValue

# Regrese

* Regrese je ve **classifiers.functions.LinearRegression**
* Pokud chci regresi polynomem vyššího stupně je to **classifiers.functions.SMOreg**
  + Zde s atributu kernel nastavuje stupen (PolyKernel -> exponent)
* Když si chci zobrazit chybu regrese, v clasify kliknu v result list na záznam pravým tlačítkem-> Visualize Classifier Errors

Kdybych chtěl vyzkoušet, který algorimus se nejvíce hodí, použiji experimenter. Zde přepnu přepínač na regresion, do datasets dám datasety s hodnotami a do algoritmů algoritmy, které mě zajímají.

Pokud chci regresí něco dopočítat, nějaké prázdné hodnoty, v exploreru si otevřu model, kde jsou všechny hodnoty, následně v classifier -> classify vyberu supplied test set dataset, pro který chceme dopočítat. Vyleze nám rovnice a s tou pak nevím coo … :D