Data Sciences with and for Data Scientists

From your analysis of a problem to a notebook and back again

**What is it?**
- Generates Notebooks with pre-existing code artifacts according to your problem and solution specifications
- Extracts knowledge from past XPS inside a supervised and monitored environment
- Reduces solution space by questioning the Data scientist

**Why are we doing it?**
- To help the Data scientist to specify her problem before leaving the machine to search for a solution
- To avoid Cargo Cult: The « best » workflow will not be the same for each problem
- To reduce the resources needed for learning today: Less meta-learning, more reuse-learning

**How does it work?**

**Challenges:**
- Respect data scientists’ practices but improve them
- Suggest additional steps (visualization, data grouping,...)
- Make adaptations easier
- Split the work into several workflows (from data analysis to tests)

**Realization:**
- The input contains a list of artifacts
- Artifacts can be data treatments, algorithms, metrics, visualization
- Artifacts are found by a mapper
- Artifacts are selected by the data scientist during the configuration process

**About the notebook generator:**

**Perspectives**

Towards notebook-related knowledge extraction:
- Solution validation
- Check reproducibility [2]
- Check problem-solution consistency
- Solution analysis
  - Check the notebook’s content
  - Check specific structural patterns
  - Check specific syntactic and semantic pattern
- Notebooks comparison
  - Match cells between original and worked notebooks
  - Compare the content on matched cells
  - Analyze content of new cells
- Solution’s knowledge extraction
  - Check the consistency of potential new version of an artifact
  - Extract new content to update artifacts
  - Store notebook and configuration

Towards supervised and monitored environment:
- Architecture requirements analysis:
  - On premise
  - ML student-friendly environment
  - Multi-users
  - Resources Monitoring
  - Research-friendly tech stack
  - Scalability
  - DevOps

**References:**