

Домашняя работа содержит следующие алгоритмы:

1. **lazy\_fca.py** - реализация простейшего алгоритма классификации из задания: объект классифицируется положительно, если каждое его пересечение с объектами из положительного контекста не вкладывается в описания из отрицательного контекста (и наоборот).
2. **fca\_votes.py** - реализация алгоритма классификации, основанного на голосовании: каждый из положительных объектов “голосует” за положительный результат, если его пересечение с тестируемым объектом не вкладывается в описания из отрицательного контекста (и наоборот).
3. **supp\_fca.py** - реализация алгоритма классификации, считающего для каждого примера его поддержку в положительном и отрицательном контексте (по формуле из задания). Выбирается класс, соответствующий контексту с большей поддержкой.
4. **supp\_falsifiab.py** - реализация алгоритма классификации, в которой из поддержки для каждого примера вычитается доля примеров, фальсифицирующих положительную/отрицательную гипотезу.
5. **antisupp\_fca.py** - реализация алгоритма классификации, считающего для каждого примера его поддержку в чужом контексте. Выбирается класс, соответствующий контексту с большей поддержкой.

	Accuracy
lazy_fca	0.2002
fca_votes	0.7838
supp_fca	0.7621
supp_falsifiab	0.7703
antisupp_fca	0.7979
SVM	0.8202

В итоге SVM оказался самым точным.