# **Table of contents**

[Introduction 1](#_Toc508323405)

[1.1. Algorithm inputs 1](#_Toc508323406)

[1.2. Algorithm output 2](#_Toc508323407)

[II. Tests 4](#_Toc508323408)

# Introduction

The Hedwig algorithm is a data mining algorithm designed for exploratory data analysis of a data set. It recieves as input a data set of instances that can have either numeric or categoric features describing them. The input instances must be labelled, meaning that each instance belongs to a given class. By default, the first column of the data set is presumed to be the label of the data instances. As output, the algorithm produces a file containing interesting patterns (in the form of rules) of the data instances, along with some quality measures of the presented patterns.

The algorithm works by first discretizing the data set. Each numeric feature is discretized into 10 approximatelly equally sized bins - because the goal of the algorithm is to produce human readable \*descriptions\* of the data set, 10 is a sensible number of bins at which interesting rules can be discovered while not making the output difficult to understand. Once the data set has been discretized, the algorithm searches for rules, describing the data set. The goal of Hedwig is to discover not simply single properties of interesting subsets of a data set, but more complex rules in the form of \*conjuncts\* of properties. For example, in the \*iris\* data set, we can discover the rule \*if 3.95<=petal\_length<5.425 \*\*and\*\* 1.3<=petal\_width<1.9, \*\*then\*\* Iris-versicolor\*. Complex rules like this are discovered using beam search, where the beam contains the best \*N\* rules found so far. It starts with the default rule which covers all the input examples. In every search iteration, each rule from the beam is specialized via one of the four operations: (1) replace the predicate of a rule with a predicate that is a sub-class of the previous one, (2) negate predicate of a rule, (3) append a new unary predicate to the rule, or (4) append a new binary predicate, introducing a new existentially quantified variable, where the new variable has to be `consumed' by a literal, which has to be added as a conjunction to this clause in the next step of rule refinement.

The algorithm has two parameters:

* *beam:* The size of the beam to be used in the search. Larger values of this variable cause the search of the algorithm to take longer and return more high quality rules.
* *support:* The minimum relative support of the rules, discovered by Hedwig. The value of this parameter must be between 0 and 1 as the parameter represents the ration of the covered examples in the entire data set.

## Algorithm inputs

In this document, we will present the results obtained on the well known iris data set. The input of the algorithm were 150 examples of plants belonging to one of three iris species. Each of the 150 examples is represented as a 4-dimensional real valued vector.

## Algorithm output

As needed by the MIP, our algorithm’s output is a tabular data resource json file. As the full output of the algorithm contains several thousand lines, we provide only the beginning of the output in this report.

|  |
| --- |
| Hedwig semantic pattern mining (anze.vavpetic@ijs.si)  Version: 0.3.1  Start: 2018-03-07T14:13:22.298356  Time taken: 0.41 seconds  Parameters:  bk\_dir=/tmp/tmpznrb6rh8  data=input.csv  format=csv  output=rules.txt  covered=None  mode=subgroups  target=None  score=lift  negations=False  alpha=0.05  adjust=fwer  FDR=0.05  leaves=True  learner=heuristic  optimalsubclass=False  uris=False  beam=20  support=0.1  depth=5  nocache=True  verbose=False  ----------------------------------------  'Iris-versicolor'(X) <--  annotated\_with(X, 0.7<=petal\_width<1.3) [cov=15, pos=15, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 0.7<=petal\_width<1.3), annotated\_with(X, root) [cov=15, pos=15, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 0.7<=petal\_width<1.3), annotated\_with(X, 0.7<=petal\_width<1.3) [cov=15, pos=15, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.2<=sepal\_length<6.1), annotated\_with(X, 1.3<=petal\_width<1.9) [cov=19, pos=16, prec=0.842, lift=2.526, pval=0.000]  annotated\_with(X, 3.95<=petal\_length<5.425), annotated\_with(X, 1.3<=petal\_width<1.9) [cov=41, pos=33, prec=0.805, lift=2.415, pval=0.000]  annotated\_with(X, 1.3<=petal\_width<1.9) [cov=51, pos=35, prec=0.686, lift=2.059, pval=0.000]  annotated\_with(X, 1.3<=petal\_width<1.9), annotated\_with(X, root) [cov=51, pos=35, prec=0.686, lift=2.059, pval=0.000]  annotated\_with(X, 1.3<=petal\_width<1.9), annotated\_with(X, 1.3<=petal\_width<1.9) [cov=51, pos=35, prec=0.686, lift=2.059, pval=0.000]  annotated\_with(X, 2.0<=sepal\_width<2.6) [cov=19, pos=13, prec=0.684, lift=2.053, pval=0.000]  annotated\_with(X, 2.0<=sepal\_width<2.6), annotated\_with(X, root) [cov=19, pos=13, prec=0.684, lift=2.053, pval=0.000]  annotated\_with(X, 2.0<=sepal\_width<2.6), annotated\_with(X, 2.0<=sepal\_width<2.6) [cov=19, pos=13, prec=0.684, lift=2.053, pval=0.000]  annotated\_with(X, 2.6<=sepal\_width<3.2), annotated\_with(X, 1.3<=petal\_width<1.9) [cov=37, pos=25, prec=0.676, lift=2.027, pval=0.000]  annotated\_with(X, 6.1<=sepal\_length<7.0), annotated\_with(X, 1.3<=petal\_width<1.9) [cov=27, pos=18, prec=0.667, lift=2.000, pval=0.000]  annotated\_with(X, 3.95<=petal\_length<5.425) [cov=61, pos=39, prec=0.639, lift=1.918, pval=0.000]  annotated\_with(X, 5.2<=sepal\_length<6.1) [cov=48, pos=26, prec=0.542, lift=1.625, pval=0.000]  annotated\_with(X, 2.6<=sepal\_width<3.2) [cov=76, pos=32, prec=0.421, lift=1.263, pval=0.000]  'Iris-virginica'(X) <--  annotated\_with(X, 1.9<=petal\_width<2.5) [cov=34, pos=34, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 1.9<=petal\_width<2.5), annotated\_with(X, root) [cov=34, pos=34, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 1.9<=petal\_width<2.5), annotated\_with(X, 1.9<=petal\_width<2.5) [cov=34, pos=34, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.425<=petal\_length<6.9) [cov=28, pos=28, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.425<=petal\_length<6.9), annotated\_with(X, root) [cov=28, pos=28, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.425<=petal\_length<6.9), annotated\_with(X, 5.425<=petal\_length<6.9) [cov=28, pos=28, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 2.6<=sepal\_width<3.2), annotated\_with(X, 1.9<=petal\_width<2.5) [cov=20, pos=20, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 6.1<=sepal\_length<7.0), annotated\_with(X, 1.9<=petal\_width<2.5) [cov=20, pos=20, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 1.9<=petal\_width<2.5), annotated\_with(X, 5.425<=petal\_length<6.9) [cov=20, pos=20, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.425<=petal\_length<6.9), annotated\_with(X, 2.6<=sepal\_width<3.2) [cov=17, pos=17, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 5.425<=petal\_length<6.9), annotated\_with(X, 6.1<=sepal\_length<7.0) [cov=16, pos=16, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 6.1<=sepal\_length<7.0) [cov=48, pos=29, prec=0.604, lift=1.812, pval=0.000]  annotated\_with(X, 6.1<=sepal\_length<7.0), annotated\_with(X, root) [cov=48, pos=29, prec=0.604, lift=1.812, pval=0.000]  annotated\_with(X, 6.1<=sepal\_length<7.0), annotated\_with(X, 2.6<=sepal\_width<3.2) [cov=32, pos=18, prec=0.562, lift=1.688, pval=0.001]  annotated\_with(X, 2.6<=sepal\_width<3.2) [cov=76, pos=32, prec=0.421, lift=1.263, pval=0.000]  'Iris-setosa'(X) <--  annotated\_with(X, 1.0<=petal\_length<2.475) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 0.1<=petal\_width<0.7) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 1.0<=petal\_length<2.475), annotated\_with(X, root) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 1.0<=petal\_length<2.475), annotated\_with(X, 0.1<=petal\_width<0.7) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 0.1<=petal\_width<0.7), annotated\_with(X, root) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 0.1<=petal\_width<0.7), annotated\_with(X, 0.1<=petal\_width<0.7) [cov=50, pos=50, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 4.3<=sepal\_length<5.2), annotated\_with(X, 0.1<=petal\_width<0.7) [cov=36, pos=36, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 3.2<=sepal\_width<3.8), annotated\_with(X, 0.1<=petal\_width<0.7) [cov=31, pos=31, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 4.3<=sepal\_length<5.2), annotated\_with(X, 3.2<=sepal\_width<3.8) [cov=23, pos=23, prec=1.000, lift=3.000, pval=0.000]  annotated\_with(X, 4.3<=sepal\_length<5.2) [cov=41, pos=36, prec=0.878, lift=2.634, pval=0.000]  annotated\_with(X, 4.3<=sepal\_length<5.2), annotated\_with(X, root) [cov=41, pos=36, prec=0.878, lift=2.634, pval=0.000]  annotated\_with(X, 3.2<=sepal\_width<3.8) [cov=49, pos=31, prec=0.633, lift=1.898, pval=0.000]  h |

# Tests

We have also prepared an integration test of the Hedwig algorithm. The test runs the algorithm on the iris set which is available on the HBP Medical Github repository. To setup the testing environment, the following docker configuration was used:

|  |
| --- |
| ---  version: '2'  services:  db:  image: postgres:9.6.5-alpine  hostname: db  environment:  POSTGRES\_PASSWORD: test  wait\_dbs:  image: "waisbrot/wait"  restart: "no"  environment:  TARGETS: "db:5432"  TIMEOUT: 60  create\_dbs:  image: "hbpmip/create-databases:1.0.0"  restart: "no"  environment:  DB\_HOST: db  DB\_PORT: 5432  DB\_ADMIN\_USER: postgres  DB\_ADMIN\_PASSWORD: test  DB1: features  USER1: features  PASSWORD1: featurespwd  DB2: woken  USER2: woken  PASSWORD2: wokenpwd  depends\_on:  - db  sample\_data\_db\_setup:  image: "hbpmip/sample-data-db-setup:0.5.0"  container\_name: "data-db-setup"  restart: "no"  environment:  FLYWAY\_DBMS: postgresql  FLYWAY\_HOST: db  FLYWAY\_PORT: 5432  FLYWAY\_DATABASE\_NAME: features  FLYWAY\_USER: postgres  FLYWAY\_PASSWORD: test  depends\_on:  - db  woken\_db\_setup:  image: "hbpmip/woken-db-setup:latest"  container\_name: "woken-db-setup"  restart: "no"  environment:  FLYWAY\_DBMS: postgresql  FLYWAY\_HOST: db  FLYWAY\_PORT: 5432  FLYWAY\_DATABASE\_NAME: woken  FLYWAY\_USER: postgres  FLYWAY\_PASSWORD: test  depends\_on:  - db  hedwig:  image: "hbpmip/python-jsi-hedwig:latest"  container\_name: "python-jsi-hedwig"  restart: "no"  environment:  FUNCTION: python-jsi-hedwig  NODE: job\_test  JOB\_ID: 2  IN\_JDBC\_DRIVER: org.postgresql.Driver  IN\_JDBC\_URL: jdbc:postgresql://db:5432/features  IN\_JDBC\_USER: features  IN\_JDBC\_PASSWORD: featurespwd  OUT\_JDBC\_DRIVER: org.postgresql.Driver  OUT\_JDBC\_URL: jdbc:postgresql://db:5432/woken  OUT\_JDBC\_USER: woken  OUT\_JDBC\_PASSWORD: wokenpwd  PARAM\_variables: "name"  PARAM\_covariables: "sepal\_length,sepal\_width,petal\_length,petal\_width"  PARAM\_grouping: ""  PARAM\_query: "SELECT name,sepal\_length,sepal\_width,petal\_length,petal\_width FROM iris"  PARAM\_meta: "{\"name\":{\"code\":\"name\",\"type\":\"string\"},\"sepal\_length\":{\"code\":\"sepal\_length\",\"type\":\"real\"},\"sepal\_width\":{\"code\":\"sepal\_width\",\"type\":\"real\"},\"petal\_length\":{\"code\":\"petal\_length\",\"type\":\"real\"}, \"petal\_width\":{\"code\":\"petal\_width\",\"type\":\"real\"}}"  PARAM\_MODEL\_beam: 5  PARAM\_MODEL\_support: 0.1  links:  - "db:db" |

When we launch the test suite, the output is as follows.

|  |
| --- |
| Starting the databases...  Creating network "tests\_default" with the default driver  Creating tests\_db\_1  Waiting for db:5432 ... up!  Everything is up  PLAY [localhost] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  TASK [Create the new database(s)"] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  changed: [localhost] => (item={'password': 'featurespwd', 'db': 'features', 'user': 'features'})  changed: [localhost] => (item={'password': u'wokenpwd', 'db': u'woken', 'user': u'woken'})  TASK [Create user(s)] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  changed: [localhost] => (item={'password': 'featurespwd', 'db': 'features', 'user': 'features'})  changed: [localhost] => (item={'password': u'wokenpwd', 'db': u'woken', 'user': u'woken'})  PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  localhost : ok=2 changed=2 unreachable=0 failed=0  Initialise the databases...  2018/03/08 14:21:01 Waiting for: tcp://db:5432  2018/03/08 14:21:02 Connected to tcp://db:5432  Flyway 4.2.0 by Boxfuse  Database: jdbc:postgresql://db:5432/features (PostgreSQL 9.6)  Successfully validated 8 migrations (execution time 00:00.052s)  Creating Metadata table: "public"."schema\_version"  Current version of schema "public": << Empty Schema >>  Migrating schema "public" to version 1.0 - create  Migrating schema "public" to version 1.1 - churn  Migrating schema "public" to version 1.2 - iris  Migrating schema "public" to version 1.3 - dummy ldsm  Migrating schema "public" to version 1.4 - dummy federation  Migrating schema "public" to version 1.5 - synthetic datasets  Migrating schema "public" with repeatable migration Create view  Migrating schema "public" with repeatable migration Setup datasets linreg\_sample,churn,iris,desd\_synth,nida\_synth,qqni\_synth  Mar 08, 2018 2:21:04 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset linreg\_sample...  Mar 08, 2018 2:21:04 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset churn...  Mar 08, 2018 2:21:04 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset iris...  Mar 08, 2018 2:21:04 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset desd\_synth...  Mar 08, 2018 2:21:05 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset nida\_synth...  Mar 08, 2018 2:21:06 PM eu.humanbrainproject.mip.migrations.R\_\_SetupValues migrate  INFO: Migrating dataset qqni\_synth...  Successfully applied 8 migrations to schema "public" (execution time 00:03.860s).  2018/03/08 14:21:07 Command finished successfully.  2018/03/08 14:21:10 Waiting for: tcp://db:5432  2018/03/08 14:21:10 Connected to tcp://db:5432  Flyway 4.2.0 by Boxfuse  Database: jdbc:postgresql://db:5432/woken (PostgreSQL 9.6)  Successfully validated 1 migration (execution time 00:00.008s)  Creating Metadata table: "public"."schema\_version"  Current version of schema "public": << Empty Schema >>  Migrating schema "public" to version 1.0 - create  Successfully applied 1 migration to schema "public" (execution time 00:00.271s).  2018/03/08 14:21:10 Command finished successfully.  Run the Hedwig algorithm...  INFO:rdflib:RDFLib Version: 4.2.2  Hedwig INFO: Starting Hedwig  INFO:Hedwig:Starting Hedwig  Hedwig INFO: Calculating data checksum  INFO:Hedwig:Calculating data checksum  Hedwig INFO: Building graph structure  INFO:Hedwig:Building graph structure  Hedwig INFO: Building the knowledge base  INFO:Hedwig:Building the knowledge base  http://kt.ijs.si/ontology/generic#6.1less\_than=sepal\_lengthless\_than7.0 48  http://kt.ijs.si/ontology/generic#5.425less\_than=petal\_lengthless\_than6.9 28  http://kt.ijs.si/ontology/generic#1.3less\_than=petal\_widthless\_than1.9 51  http://kt.ijs.si/ontology/generic#2.0less\_than=sepal\_widthless\_than2.6 19  http://kt.ijs.si/ontology/generic#0.1less\_than=petal\_widthless\_than0.7 50  http://kt.ijs.si/ontology/generic#3.8less\_than=sepal\_widthless\_than4.4 6  http://kt.ijs.si/ontology/generic#5.2less\_than=sepal\_lengthless\_than6.1 48  http://kt.ijs.si/ontology/generic#1.0less\_than=petal\_lengthless\_than2.475 50  http://kt.ijs.si/ontology/generic#3.2less\_than=sepal\_widthless\_than3.8 49  http://kt.ijs.si/ontology/generic#2.475less\_than=petal\_lengthless\_than3.95 11  http://kt.ijs.si/ontology/generic#0.7less\_than=petal\_widthless\_than1.3 15  http://kt.ijs.si/ontology/generic#4.3less\_than=sepal\_lengthless\_than5.2 41  http://kt.ijs.si/ontology/generic#3.95less\_than=petal\_lengthless\_than5.425 61  http://kt.ijs.si/ontology/generic#2.6less\_than=sepal\_widthless\_than3.2 76  http://kt.ijs.si/ontology/generic#1.9less\_than=petal\_widthless\_than2.5 34  http://kt.ijs.si/ontology/generic#7.0less\_than=sepal\_lengthless\_than7.9 13  Hedwig INFO: Starting learner for target 'Iris-versicolor'  INFO:Hedwig:Starting learner for target 'Iris-versicolor'  Hedwig INFO: Validating rules, alpha = 0.050  INFO:Hedwig:Validating rules, alpha = 0.050  Hedwig INFO: Starting learner for target 'Iris-setosa'  INFO:Hedwig:Starting learner for target 'Iris-setosa'  Hedwig INFO: Validating rules, alpha = 0.050  INFO:Hedwig:Validating rules, alpha = 0.050  Hedwig INFO: Starting learner for target 'Iris-virginica'  INFO:Hedwig:Starting learner for target 'Iris-virginica'  Hedwig INFO: Validating rules, alpha = 0.050  INFO:Hedwig:Validating rules, alpha = 0.050  Hedwig INFO: Finished in 0 seconds  INFO:Hedwig:Finished in 0 seconds  Hedwig INFO: Outputing results  INFO:Hedwig:Outputing results  Stopping the containers...  Stopping tests\_db\_1 ... done  Removing tests\_hedwig\_run\_1 ... done  Removing tests\_woken\_db\_setup\_run\_1 ... done  Removing tests\_sample\_data\_db\_setup\_run\_1 ... done  Removing tests\_create\_dbs\_run\_1 ... done  Removing tests\_wait\_dbs\_run\_1 ... done  Removing tests\_db\_1 ... done  Removing network tests\_default  Stopping the containers...  Removing network tests\_default  WARNING: Network tests\_default not found. |