Index

0 – 1 loss function, 160
0.632 bootstrap, 155
1R (1-rule), 86–90
  discretization, 315
  example use, 87t
  learning procedure, 89–90
  missing values and numeric data, 87–89
  overfitting for, 88
  pseudocode, 86f
11-point average recall, 175

A

accuracy, of association rules, 72, 73, 116
  minimum, 72, 119, 122–123
accuracy, of classification rules, 110, 205
activation function, 241–242
acuity parameter, 281
AD trees. *See* all-dimensions trees
AdaBoost, 358–361
AdaBoostM1 algorithm, 358–359, 475t, 476–477
Add filter, 433t–435t, 436
AddClassification filter, 444t, 445
AddCluster filter, 433t–435t, 436–437
AddExpression filter, 433t–435t, 437
AddID filter, 433t–435t, 436
additive logistic regression, 364–365
additive regression, 362–365
AdditiveRegression algorithm, 475t, 476
AddNoise filter, 433t–435t, 441, 568
AddValues filter, 433t–435t, 438
ADTree algorithm, 446t–450t, 457
adversarial data mining, 393–395
agglomerative clustering, 273, 275–276
Akaike Information Criterion (AIC), 267, 456
algorithms. *See specific Weka algorithms*
all-dimensions (AD) trees, 270–271
  generation, 271–272
  illustrated examples, 271f
alternating decision trees, 366–367
  example, 367, 367f
  prediction nodes, 366–367
  splitter nodes, 366–367
Analyze panel, 505–509, 512–515
ancestor-of relation, 46
AND, 233
anomalies, detecting, 334–335
antecedent, of rule, 67, 69
AODE. *See* averaged one-dependence estimator
AODE algorithm, 446t–450t, 451
AODEsr algorithm, 446t–450t, 451
applications, 375–399
  automation, 28
  challenge of, 375
  data stream learning, 380–383
  diagnosis, 25–26
  fielded, 21–28
  incorporating domain knowledge, 384–386
  massive datasets, 378–380
  message classifier, 531–538
  text mining, 386–389
Apriori algorithm, 216
Apriori rule learner, 485–486, 486t
  default options, 582
  output for association rules, 430f
  parameters, 584
area under the curve (AUC), 177, 580
area under the precision-recall curve (AUPRC), 177
ARFF files, 52–56
  attribute specifications in, 54
  attribute types in, 54
  converting files to, 417–419
  defined, 52–56
  illustrated, 53f
  in Weka, 407
arithmetic underflow, 266–267
assignment of key phrases, 387–388
association learning, 40
association rule mining, 582–584
association rules, 11, 72–73. *See also* rules
  accuracy (confidence), 72–73, 116
  characteristics, 72
  computation requirement, 123–124
  converting item sets to, 119
  coverage (support), 72, 116
  double-consequent, 123
  examples, 11
  finding, 116
  finding large item sets, 219–222
  frequent-pattern tree, 216–219
  mining, 116–124
  predicting multiple consequences, 72
  relationships between, 73

Note: Page numbers followed by "f" indicates a figure, "t" indicates a table, and "b" indicates entry is inside boxed text.

association rules *(continued)*
single-consequent, 123
in Weka, 429–430
association-rule learners, 485–487
attribute evaluation methods, 487–494
attribute subset evaluators, 488
list of, 489t
single-attribute evaluators, 490–492
attribute filters, 432
supervised, 443–445
unsupervised, 432–441
attribute selection, 306, 307–314. *See also* data transformation
backward elimination, 311–312
beam search, 312
best-first search, 312
filter method, 308–309
forward selection, 311–312
instance-based learning methods, 310
nearest-neighbor learning, 567–568
race search, 313
recursive feature elimination, 309–310
schemata search, 313
scheme-independent, 308–310
scheme-specific, 312–314
searching the attribute space and, 311–312
selective Naïve Bayes, 314
symmetric uncertainty, 310b
in Weka, 430
Weka evaluation methods for, 487–494
Weka Explorer exercise, 575–577
Weka search methods for, 490t
wrapper method, 308–309
attribute subset evaluators, 488
attribute-efficient learners, 131
attributes, 9–10, 39, 49
adding, 436–438
ARFF format, 54
Boolean, 51
causal relations, 384
combination of, 116
conversions, 438–439
date, 54
difference, 132
discrete, 51
evaluating, 87t
highly branching, 105–107
identification code, 88
interval, 50
irrelevant, 308
nominal, 49, 289
normalized, 57
numeric, 49, 193–194
ordinal, 50–51
ratio, 50
relations between, 77
relation-valued, 54–55
relevant, 308
removing, 436–438
semantic relation between, 384
string, 54, 579–580
string, conversion, 439–440
types of, 39, 56–58
values of, 49
values of, changing, 438
weighting, 246–247
AttributeSelectedClassifier algorithm, 475t, 478, 582
AttributeSelection filter, 444, 444t
AttributeSummarizer, 498, 499t
AUC. *See* area under the curve
AUPRC. *See* area under the precision-recall curve
authorship ascription, 387–388
AutoClass, 273, 291, 293
automatic attribute selection, 562, 575–576
automatic parameter tuning, 577–578
automation applications, 28
averaged one-dependence estimator (AODE), 269, 451
average-linkage method, 275–276

B

background knowledge, 380
backpropagation, 235–241
stochastic, 238b–239b
backward elimination, 311–312
backward pruning, 195
bagging, 352–356
algorithm for, 355f
bias-variance decomposition, 353–355
with costs, 355–356
idealized procedure versus, 354
instability neutralization, 354
for numeric prediction, 354–355
as parallel, 379
randomization versus, 357
in Weka, 474–479
Bagging algorithm, 474, 475t
bags, 141–142
class labels, 142–143
instances, joining, 300
positive, 301–302
positive probability, 302
balanced iterative reducing and clustering using hierarchies (BIRCH), 293

Balanced Winnow, 131
ball trees, 135
  in finding nearest neighbors, 136
  illustrated, 136f
  nodes, 135–136
  splitting method, 136–137
  two cluster centers, 141f
batch learning, 238b–239b
Bayes Information Criterion, 293
Bayesian clustering, 290–292
  AutoClass, 291
  DensiTree, 292, 292f
  hierarchical, 292
Bayesian multinet, 270
Bayesian networks, 143, 261–273
  algorithms, 268–270
  AD tree, 270–272, 271f
  conditional independence, 264–266
  data structures for fast learning, 270–272
  example illustrations, 263f, 265f
  K2 algorithm, 273
  learning, 266–268
  making predictions, 262–266
  Markov blanket, 269
  prior distribution over network structures, 268
  structure learning by conditional independence tests, 270
  TAN, 269
  visualization example, 454f
BayesianLogisticRegression algorithm, 446t–450t, 453
BayesNet algorithm, 446t–450t, 453
beam search, 312
Bernoulli process, 150
BestFirst method, 490t, 492
best-first search, 312
BFTree algorithm, 446t–450t, 456
bias, 31–33
  language, 31–32
  multilayer perceptron, 233
  overfitting-avoidance, 32–33
  search, 32
bias-variance decomposition, 353–355
binary classification problems, 63
BIRCH. *See* balanced iterative reducing and clustering using hierarchies
bits, 100–101
Boolean attributes, 51
Boolean classes, 71–72
boosting, 358–362
  AdaBoost, 358–361
  algorithm for, 359–360, 359f
  classifiers, 362
  in computational learning theory, 361
  decision stumps, 362
  forward stagewise additive modeling, 362
  power of, 361–362
  in Weka, 476–477
bootstrap, 155–156
bootstrap aggregating. *See* bagging
Boundary Visualizer, 571, 574
buildClassifier() method, 537–538, 540, 555

C

C4.5, 108, 198b, 201–202, 307–308
  functioning of, 201
  MDL-based adjustment, 201–202
C5.0, 254–255
calibration, class probability, 343–346
  discretization-based, 345
  logistic regression, 346
  PAV-based, 345–346
Capabilities class, 540, 556–557
CAPPS. *See* Computer-Assisted Passenger Prescreening System
CART system, 192, 261, 456
  cost-complexity pruning, 202
categorical attributes. *See* nominal attributes
category utility, 273, 284–285
  calculation, 284b–285b
  incremental clustering, 279, 281
causal relations, 384
CBA technique, 223
Center filter, 433t–435t, 437
CfsSubsetEval method, 488, 489t
chain rule, 342–343
ChangeDateFormat filter, 433t–435t, 440
ChiSquaredAttributeEval, 489t, 491
circular ordering, 51
CitationKNN algorithm, 446t–450t, 473
class boundaries
  non-axis parallel, 250
  rectangular, 248, 249f
class labels
  bags, 142–143
  reliability, 377–378
class noise, 568
class probability estimation, 337
  dataset with two classes, 344, 344f
  difficulty, 343–344
  overoptimistic, 344
ClassAssigner component, 495, 499–500, 499t
ClassAssigner filter, 433t–435t, 438

classes, 40
Boolean, 71–72
membership functions for, 125
rectangular, 248, 249f
in Weka, 520
classification, 40
clustering for, 294–296
cost-sensitive, 166–167, 356
document, 387–388
k-nearest-neighbor, 78
Naïve Bayes for, 97–98
nearest-neighbor, 78
one-class, 335
pairwise, 339
classification boundaries, 571–574
1R visualization, 571–572
decision tree visualization, 573
Naïve Bayes visualization, 573
nearest-neighbor visualization, 572
rule sets visualization, 573
classification learning, 40
classification rules, 11, 62. *See also* rules
accuracy, 205
antecedent of, 69
criteria for choosing tests, 203–204
disjunctive normal form, 71–72
with exceptions, 73, 194
exclusive-or, 70, 70f
global optimization, 208
good rule generation, 205–208
missing values, 204–205
multiple, 71
numeric attributes, 205
from partial decision trees, 208–212
producing with covering algorithms, 205
pruning, 206
replicated subtree, 69, 71f
RIPPER rule learner, 208, 209f, 215
ClassificationViaClustering algorithm, 475t, 479
ClassificationViaRegression algorithm, 475t, 479
Classifier class, 539, 549, 553–555
ClassifierPerformanceEvaluator, 495–496, 499–502, 499t
classifiers package, 519–520
classifiers (Weka), 526
capabilities, 555–557
implementation conventions, 555–557
ClassifierSubsetEval method, 488, 489t, 493
Classify panel, 422–424, 562–565
with C4.5 algorithm, 562–563
classification error visualization, 565
output interpretation, 563–564
setting test method, 565
classifyInstance() method, 549–550
classifyMessage() method, 537–538
ClassOrder filter, 444, 444t
ClassValuePicker, 499–500, 499t
CLI. *See* command-line interface
CLOPE algorithm, 480t, 483
closed-world assumptions, 43, 71–72
CLOSET+ algorithm, 223
ClustererPerformanceEvaluator, 499–500, 499t
clustering, 40, 89
agglomerative, 273, 275–276
algorithms, 81, 89
Bayesian, 290–292
category utility, 273
for classification, 294–296
document, 387
EM algorithm, 287
evaluation, 186
group-average, 276
in grouping items, 41
hierarchical, 274–279
incremental, 279–284
iterative distance-based, 139
k-means, 139–140
MDL principle application to, 186–187
number of clusters, 274
probability-based, 285–286
representation, 82f
stage following, 81
statistical, 314–315
in Weka, 429
Weka algorithms, 480–485
ClusterMembership filter, 433t–435t, 436–437
Cobweb algorithm, 429, 480, 480t, 483
co-EM, 297
column separation, 340–341
combining classifiers, in Weka, 477
command-line interface (CLI), 519–530. *See also* Weka
generic options, 526–529
options, 526–529
packages, 519
scheme-specific options, 528t, 529
starting up, 519
weka.associations package, 525
weka.attributeSelection package, 525
weka.classifiers package, 523–525
weka.clusterers package, 525
weka.core package, 520–523
weka.datagenerators package, 525

weka.estimators package, 525
weka.filters package, 525
comma-separated value (CSV)
data files, 408
example data, 409f
format, 407
ComplementNaiveBayes algorithm, 446t–450t
complete-linkage method, 275
computational learning theory, 361
computeInfoGain() method, 549
Computer-Assisted Passenger Prescreening System (CAPPS), 394
concept descriptions, 39–40
concepts, 40–42. *See also* input
defined, 39
conditional independence, 264–266
confidence
of association rules, 72, 116
intervals, 150
upper/lower bounds, 246
confidence limits
in error rate estimation, 197–198
for normal distribution, 152t
on success probability, 246
for Student's distribution, 159t
confusion matrix, 164
ConjunctiveRule algorithm, 446t–450t, 459
consequent, of rule, 67
ConsistencySubsetEval method, 488, 489t
constrained quadratic optimization, 225
constructors, 523–524
contact lens problem, 12–13
covering algorithm, 110–115
rules, 12f
structural description, 13, 13f
continuous attributes. *See* numeric attributes
convex hulls, 224–225
Copy filter, 433t–435t, 436–437
corrected resampled *t*-test, 159
cost curves, 177–180
cost in, 179
cost matrixes, 166–167, 166t, 172
cost of errors, 163–180
cost curves, 177–180
cost-sensitive classification, 166–167
cost-sensitive learning, 167–168
examples, 163–164
lift charts, 168–172
problem misidentification, 163–164
recall-precision curves, 174–175
ROC curves, 172–174
cost–benefit analyzer, 170
CostBenefitAnalysis, 498, 499t
cost-complexity pruning, 202
cost-sensitive classification, 166–167, 356
in Weka, 477
cost-sensitive learning, 167–168
two-class, 167–168
in Weka, 477
CostSensitiveAttributeEval method, 488, 489t, 491–492
CostSensitiveClassifier algorithm, 475t, 477–478
CostSensitiveSubsetEval method, 489t
co-training, 296
EM and, 297
counting the cost, 163–180
covariance matrix, 289
coverage, of association rules, 72, 116
dividing, 122–123
minimum, 122–123
specifying, 123–124
covering algorithms, 108–116
example, 110–115
illustrated, 109f
instance space during operation of, 110f
operation, 110
in producing rules, 205
in two-dimensional space, 108
CPU performance, 15
dataset, 16t
in Weka, 423f
cross-validation, 89, 152–154
estimates, 157–159
folds, 153
leave-one-out, 154
repeated, 159
for ROC curve generation, 173
stratified threefold, 153
tenfold, 153–154, 306
threefold, 153
CrossValidationFoldMaker, 495–496, 499–502, 499t
CSV. *See* comma-separated value
CSVLoader, 417–418
cumulative margin distribution, 528–529
customer support/service applications, 28
cutoff parameter, 283
CVParameterSelection algorithm, 475t, 478, 578

D

Dagging algorithm, 474–476, 475t
data, 35
  linearly separable, 127–128
  noise, 6–7
  overlay, 52
  scarcity of, 397
  sparse, 56
  with string attributes, 579–580
data cleansing, 60, 307, 331–337. *See also* data transformation
  anomaly detection, 334–335
  decision tree improvement, 332
  methods, 307
  one-class learning, 335–337
  robust regression, 333–334
data mining, 4–5, 8–9
  adversarial, 393–395
  applying, 375–378
  as data analysis, 4–5
  ethics and, 33–36
  learning machine and, 3–9
  scheme comparison, 156–157
  ubiquitous, 395–397
data preparation. *See also* input
  ARFF files, 52–56
  attribute types, 56–58
  data gathering in, 51–52
  data knowledge and, 60
  inaccurate values in, 59–60
  missing values in, 58–59
  sparse data, 56
data projections, 306–307, 322–330
  partial least-squares regression, 326–328
  principal components analysis, 324–326
  random, 326
  text to attribute vectors, 328–329
  time series, 330
data stream learning, 380–383
  algorithm adaptation for, 381–382
  Hoeffding bound, 382
  memory usage, 383
  Naïve Bayes for, 381
  tie-breaking strategy, 383
data transformations, 305–349
  attribute selection, 306–314
  data cleansing, 307, 331–337
  data projection, 306–307, 322–330
  discretization of numeric attributes, 306, 314–322
  input types and, 323
  methods for, 306
  multiple classes to binary ones, 307, 338–343
  sampling, 307, 330–331
data warehousing, 52
dataSet connections, 501
DataVisualizer, 498, 499t
date attributes, 54
DBScan algorithm, 480t, 483–485, 484f
decision boundaries, 63
decision lists, 10
  rules versus, 115–116
decision stumps, 362
decision tree induction, 29, 332
  complexity, 199–200
  top-down, 202–203
decision trees, 5, 64, 103f
  alternating, 366–368, 367f
  C4.5 algorithm and, 201–202
  constructing, 99–108
  cost-complexity pruning, 202
  for disjunction, 69f
  error rate estimation, 197–198
  examples, 13f, 18f
  highly branching attributes, 105–107
  improving, 332
  information calculation, 103–104
  interactive construction, 569–571
  missing values, 64, 194–195
  multivariate, 203
  nodes, 64
  numeric attributes, 193–194
  partial, obtaining rules from, 208–212
  pruning, 195–197
  with replicated subtree, 71f
  rules, 200–201
  top-down induction of, 107–108
  univariate, 203
  visualizing, 573
  in Weka, 410–414
  Weka Explorer exercise, 566–571
DecisionStump algorithm, 446t–450t, 455
DecisionTable algorithm, 446t–450t, 457
Decorate algorithm, 475t, 476
dedicated multi-instance methods, 301–302
Delta, 330
dendrograms, 81, 274–275
denormalization, 44
  problems with, 46
DensiTree, 292, 292f
diagnosis applications, 25–26
  faults, 25–26
  machine language in, 25
  performance tests, 26

difference attributes, 132
direct marketing, 27
directed acyclic graphs, 262
discrete attributes, 51
  converting to numeric attributes, 322
  discretization, 306, 314–322. *See also* data transformation
  1R (1-rule), 315
  decision tree learners, 315
  entropy-based, 316–319
  error-based, 320–322
  global, 315
  partitioning, 87
  proportional *k*-interval, 316
  supervised, 316, 574
  unsupervised, 316, 574
  Weka Explorer exercise, 574–575
  Weka metalearner for, 443f
discretization-based calibration, 345
Discretize filter, 416, 433t–435t, 438, 444t
disjunctive normal form, 71–72
distance functions, 131–132
  difference attributes, 132
  generalized, 249–250
  for generalized exemplars, 248–249
  missing values, 132
distribution, in Weka, 515–517
diverse-density method, 302
divide-and-conquer, 99–108, 308
DMNBText algorithm, 446t–450t, 453
document classification, 387. *See also* classification
  actual documents, 580–581
  in assignment of key phrases, 387–388
  in authorship ascription, 387–388
  data with string attributes, 579–580
  in language identification, 387–388
  as supervised learning, 387
  Weka Explorer exercise, 578–582
document clustering, 387
domain knowledge, 19
double-consequent rules, 123
DTNB algorithm, 446t–450t, 457

E

early stopping, 238b–239b
eigenvalues, 324
eigenvectors, 324
EM algorithm, 480–483, 480t, 482f
embedded machine learning, 531–538
END algorithm, 475t
ensemble learning, 351–373
  additive regression, 362–365
  bagging, 352–356
  boosting, 358–362
  interpretable ensembles, 365–369
  multiple models, 351–352
  randomization, 356–358
  stacking, 369–371
entity extraction, in text mining, 388
entropy, 104
entropy-based discretization, 316–319
  error-based discretization versus, 320–322
  illustrated, 318f
  with MDL stopping criterion, 320
  results, 318f
  stopping criteria, 315, 318–319
enumerated, 51
enumerating concept space, 30–31
equal-frequency binning, 316
equal-interval binning, 316
error log, 415–416
error rate, 148
  decision tree, 197–198
  repeated holdout, 152–153
  success rate and, 197–198
  training set, 148
error-based discretization, 320–322
errors
  classification, visualizing, 565
  estimation, 156
  inaccurate values and, 59–60
  mean-absolute, 181
  mean-squared, 181
  propagation, 238b–239b
  relative-absolute, 181
  relative-squared, 181
  resubstitution, 148–149
  squared, 161
  training set, 197
estimation error, 156
ethics, 33–36
  issues, 35–36
  personal information and, 34–35
  reidentification and, 33–34
Euclidean distance, 131
  function, 246
  between instances, 276
evaluation
  clustering, 186
  as data mining key, 147
  numeric prediction, 180–182
  performance, 148
examples, 42–49. *See also* instances; *specific examples*
  class of, 40
  relations, 43–46

examples *(continued)*
structured, 46–49
types of, 43–49
exceptions, rules with, 73–75, 212–215
exclusive-or problem, 70f
exclusive-OR (XOR), 233
exemplars, 245
generalizing, 247–249
noisy, pruning, 245–246
reducing number of, 245
exhaustive error-correcting codes, 341
ExhaustiveSearch method, 490t, 493
expectation, 289
expectation maximization (EM) algorithm, 287
maximization step, 295–296
with Naïve Bayes, 295
Experimenter, 405, 505–517. *See also* Weka
advanced setup, 511–512
Analyze panel, 505–509, 512–515
distributed processing, 515–517
experiment illustration, 506f–508f
results analysis, 509–510
Run panel, 505–506
running experiments, 505
Setup panel, 505, 510
simple setup, 510–511
starting up, 505–510
expert models, 352
Explorer, 404, 407–494. *See also* Weka
applying filters, 561
ARFF format, 417–419
Associate panel, 429–430
association-rule learning, 485–487
attribute selection, 487–494
automatic attribute selection, 562, 575–576
automatic parameter tuning, 577–578
classification boundaries, 571–574
Classify panel, 422–424, 562–565
Cluster panel, 429
clustering algorithms, 480–485
CSV data files, 408
CSVLoader, 417–418
Data Visualizer, 427
decision tree building, 410–414, 566–571
discretization, 574–575
document classification, 578–582
error log, 415–416
filtering algorithms, 432–445
filters, 419–422
interface illustration, 408f
introduction to, 559–565
J4.8, 410–414
learning algorithms, 445–474
loading and filtering files, 416–422
loading data into, 408–410
loading datasets, 559–560
market basket analysis, 584–585
metalearners, 427
metalearning algorithms, 474–479
models, 414–415
nearest-neighbor learning, 566–571
neural networks, 469–472
Preprocess panel, 411, 416, 419, 561
preprocessing, 574–578
real-world dataset mining, 584
search methods, 492–494
Select Attributes panel, 430, 478
training/testing learning schemes, 422–424
Tree Visualizer, 427
tutorial exercises for, 559–585
User Classifier, 424–427
Viewer, 560, 560f
Visualize panel, 430–432, 562
eXtensible Markup Language (XML), 52–56

F

false negatives (FN), 164, 176t, 580
false positive rate, 164
false positives (FP), 164, 176t, 580
Familiar system, 396–397
FarthestFirst algorithm, 480t, 483
FastVector, 536
feature selection, 346
feed-forward networks, 238b–239b
fielded applications, 21–28
automation, 28
customer service/support, 28
decisions involving judgments, 22–23
diagnosis, 25–26
image screening, 23–24
load forecasting, 24–25
manufacturing processes, 27
marketing and sales, 26–27
scientific, 28
web mining, 5
fields, 525
file mining, 48–49
files
ARFF, 52–56
filtering, 419–422

loading, 416–422
opening, 416
filter method, 308–309
FilteredAssociator rule learner, 486t, 487
FilteredAttributeEval method, 489t, 491–492
FilteredClassifier algorithm, 475t, 569
FilteredClassifier metalearning scheme, 443–444, 538
FilteredCluster algorithm, 480t, 483
FilteredSubsetEval method, 488, 489t
filtering algorithms (Weka), 432–445
filtering approaches, 334–335
filters, 404
applying, 421
applying (Weka Explorer), 561
attribute, 432–441, 443–445
choosing, 420f
information on, 421
instance, 432, 441–442, 445
supervised, 432, 443–445
unsupervised, 432–442
in Weka, 411
finite mixtures, 286
FirstOrder filter, 433t–435t, 439
fixed set, 492
fixed width, 492
flat files, 42
F-measure, 175, 479
forward pruning, 195
forward selection, 311–312
forward stagewise additive modeling, 362
implementation, 363
numeric prediction, 362–363
overfitting and, 363
residuals, 368–369
FP-growth algorithm, 216, 223
FPGrowth rule learner, 486–487, 486t
frequent-pattern trees, 216
building, 216–219
compact structure, 216–217
data preparation example, 217t–218t
header tables, 219–222
implementation, 222–223
speed, 222
structure illustration, 220f–221f
support threshold, 222–223
FT algorithm, 446t–450t, 456–457
functional dependencies, 385
functional trees, 65
functions, Weka algorithms, 446t–450t, 459–469

G

gain ratio, 105–107
GainRatioAttributeEval method, 489t, 491
Gaussian process regression, 243
GaussianProcesses algorithm, 446t–450t, 464
generalization
exemplar, 247–249, 251
instance-based learning and, 251
stacked, 369–371
generalization as search, 29
bias, 31–33
enumerating the concept space, 30–31
generalized distance functions, 249–250
Generalized Sequential Patterns (GSP), 223
GeneralizedSequentialPatterns rule learner, 486t, 487
generalizing exemplars, 247–248
distance functions for, 248–249
nested, 248
generic options (CLI), 526–529
list of, 527t
GeneticSearch method, 490t, 493
getCapabilities() method, 539
getTechnicalInformation() method, 539
glass dataset, 566–567
global optimization, classification rules for, 208
globalInfo() method, 539
gradient ascent, 302
gradient descent, 238b–239b
illustrated, 237f
stochastic, 242–243
subgradients, 242
Grading algorithm, 475t, 477
GraphViewer, 498, 499t
greedy method, for rule pruning, 253–254
GreedyStepwise method, 490t, 492–493
GridSearch algorithm, 475t, 478
group-average clustering, 276
growing sets, 205–206
GSP. *See* Generalized Sequential Patterns

H

Hamming distance, 339–340
Hausdorff distance, 301, 303
hidden layer, multilayer perceptrons, 233, 238b–239b, 239f
hierarchical clustering, 274–276. *See also* clustering
agglomerative, 275–276
average-linkage method, 275–276
centroid-linkage method, 275
dendrograms, 274–275
displays, 277f–278f

hierarchical clustering *(continued)*
example, 276–279
example illustration, 282f–283f
group-average, 276
single-linkage algorithm, 275, 279
HierarchicalClusterer algorithm, 480t, 483
highly branching attributes, 105–107
hinge loss, 242–243, 242f
histogram equalization, 316
HNB algorithm, 446t–450t, 451
Hoeffding bound, 382
Hoeffding trees, 382–383
HTML. *See* HyperText Markup Language
hyperpipes, 143
HyperPipes algorithm, 446t–450t, 474
hyperplanes, 127
maximum-margin, 224–225
separating classes, 225b
hyperrectangles, 247
boundaries, 247
exception, 248
measuring distance to, 249
in multi-instance learning, 303
overlapping, 248
hyperspheres, 135
HyperText Markup Language (HTML)
delimiters, 390
formatting commands, 389–390

I

IB1 algorithm, 446t–450t, 472
IB3. *See* Instance-Based Learner version 3
IBk algorithm, 446t–450t, 472
Id3 algorithm, 446t–450t
ID3 decision tree learner, 107–108, 539–555
buildClassifier() method, 540
classifyInstance() method, 549–550
computeInfoGain() method, 549
gain ratio, 107–108
getCapabilities() method, 539
getTechnicalInformation() method, 539
globalInfo() method, 539
improvements, 108
main() method, 553–555
makeTree() method, 540–549
Sourcable interface, 539, 550
source code, 541f–548f
TechnicalInformationHandler interface, 539
toSource() method, 550–553
identification code attributes, 88
example, 106t
image screening, 23–24
hazard detection system, 23
input, 23
problems, 24
implementations (real machine learning schemes), 191–304
association rules, 216–223
Bayesian networks, 261–273
classification rules, 203–216
clustering, 273–293
decision trees, 192–203
instance-based learning, 244–251
linear model extension, 223–244
multi-instance learning, 298–303
numeric prediction with local linear models, 251–261
semisupervised learning, 294–298
inaccurate values, 59–60
incremental clustering, 279–284
acuity parameter, 281
category utility, 279, 281
cutoff parameter, 283
example illustrations, 280f, 282f–283f
merging, 281
splitting, 281
incremental learning, 502–503
incremental reduced-error pruning, 206, 207f
IncrementalClassifierEvaluator, 498–500, 499t
inductive logic programming, 77
InfoGainAttributeEval method, 489t, 491, 582
information, 35, 100–101
calculating, 103–104
extraction, 388–389
gain calculation, 203–204
measure, 103–104
value, 104
informational loss function, 161–163
information-based heuristics, 204
input, 39–60
aggregating, 142
ARFF format, 52–56
attribute types, 56–58
attributes, 39
concepts, 40–42
data assembly, 51–52
data transformations and, 323
examples, 42–49
flat files, 42–43
forms, 39
inaccurate values, 59–60
instances, 42–49
missing values, 58–59
preparing, 51–60

sparse data, 56
tabular format, 124
input layer, multilayer perceptrons, 233
instance connections, 193
instance filters, 432
supervised, 445
unsupervised, 441–442
instance space
in covering algorithm operation, 110f
partitioning methods, 80f
rectangular generalizations in, 79
Instance-Based Learner version 3 (IB3), 246
instance-based learning, 78, 131–138
in attribute selection, 310
characteristics, 78
distance functions, 131–132
distance functions for generalized exemplars, 200
explicit knowledge representation and, 250–251
generalization and, 251
generalizing exemplars, 247–248
nearest-neighbor, 132–137
performance, 246
pruning noise exemplars, 245–246
reducing number of exemplars, 245
visualizing, 81
weighting attributes, 246–247
instance-based representation, 78–81
instances, 9–10, 39, 42
centroid, 139
misclassified, 128–130
with missing values, 194
multilabeled, 40
order, 55
sparse, 442
subset sort order, 194
training, 184
in Weka, 520
InstanceStreamToBatchMaker, 499–500, 499t
interactive decision tree construction, 569–571
interpretable ensembles, 365–369
logistic model trees, 368–369
option trees, 365–368
InterquartileRange filter, 433t–435t, 436
interval quantities, 50
iris example, 13–15
boundary decision, 63, 63f
data as clustering problem, 41t
dataset, 14t
DBScan clusters, 484f
decision tree, 65, 66f
hierarchical clusterings, 282f–283f
incremental clustering, 279–284
Logistic output, 468f
OPTICS visualization, 485f
rules, 14
rules with exceptions, 73–75, 74f, 213–215, 213f
SMO output, 463f–464f
SMO output with nonlinear kernel, 465f–467f
visualization, 431f
isotonic regression, 345
IsotonicRegression algorithm, 446t–450t, 462
item sets, 116
checking, of two consecutive sizes, 123
converting to rules, 119
in efficient rule generation, 122–123
example, 117t–118t
large, finding with association rules, 219–222
minimum coverage, 122
subsets of, 122–123
items, 116
iterative distance-based clustering, 139

J

J48 algorithm, 410–411, 446t–450t, 498, 502–503, 505, 519
changing parameters for, 455f
cross-validation with, 498–500
discretization and, 575
evaluation method, 413
output, 412f
parentheses following rule, 459
result visualization, 415f
using, 411f
J48graft algorithm, 446t–450t, 455
Java database connectivity (JDBC)
databases, 515
drivers, 419, 510–511, 515
Java virtual machine, 519
Javadoc indexes, 525–526
JRip algorithm, 446t–450t, 459
judgment decisions, 22–23

K

K2 algorithm, 273
Kappa statistic, 166, 413
kD-trees, 132
building, 133
in finding nearest-neighbor, 133–134, 134f
for training instances, 133f
updating, 135
kernel logistic regression, 231–232

kernel perceptron, 231–232
kernel ridge regression, 229–231
  computational expense, 230b
  computational simplicity, 230b
  drawback, 230b
kernel trick, 229–230
KernelFilter filter, 433t–435t, 439
k-means clustering, 139
  iterations, 139–140
  k-means++, 139
  seeds, 139
k-nearest-neighbor method, 78
knowledge, 35
  background, 380
  metadata, 385
  prior domain, 385
Knowledge Flow interface, 404–405, 495–503. *See also* Weka
  Associations panel, 498
  Classifiers panel, 498
  Clusters panel, 498
  components, 498–500
  components configuration and connection, 500–502
  dataSet connections, 501
  evaluation components, 498–500, 499t
  Evaluation panel, 495–496, 499–500
  Filters panel, 498
  illustrated, 496f
  incremental learning, 502–503
  instance connections, 193
  operations, 500f
  starting up, 495–498
  visualization components, 498–500, 499t
knowledge representation, 85–145
  clusters, 81
  instance-based, 78–81
  linear models, 62–63
  rules, 67–77
  tables, 61–62
  trees, 64–67
KStar algorithm, 446t–450t, 472
Kullback-Leibler distance, 473

L

labor negotiations example, 15–19
  dataset, 17t
  decision trees, 18f
  OneR output, 458f
  PART output, 460f–461f
  training dataset, 18–19
LADTree algorithm, 446t–450t, 457
language bias, 31–32
language identification, 387–388
Laplace estimator, 93, 291
large item sets, finding with association rules, 219–222
LatentSemanticAnalysis method, 489t, 491
LaTeX typesetting system, 514–515
law of diminishing returns, 379
lazy classifiers, in Weka, 446t–450t, 472
LBR algorithm, 446t–450t, 472
learning
  association, 40
  batch, 238b–239b
  classification, 40
  concept, 8
  cost-sensitive, 167–168
  data stream, 380–383
  ensemble, 351–373
  incremental, 502–503
  instance-based, 78, 131–138, 244–251
  locally weighted, 259–261
  machine, 7–8
  multi-instance, 48, 141–143, 298–303
  one-class, 307, 335–337
  in performance situations, 21
  rote, 78
  semisupervised, 294–298
  statistics versus, 28–29
  testing, 7
  training/testing schemes, 422–424
learning algorithms, 445–474
  Bayes, 446t–450t, 451–453
  functions, 446t–450t, 459–469
  lazy, 410–411, 446t–450t
  MI, 446t–450t, 472–474
  miscellaneous, 446t–450t, 474
  neural networks, 469–472
  rules, 446t–450t, 457–459
  trees, 446t–450t, 454–457
learning Bayesian networks, 266–268
learning paradigms, 380
learning rate, 238b–239b
learning scheme creation, in Weka, 539–557
least-squares linear regression, 63, 125–126
LeastMedSq algorithm, 446t–450t, 462
leave-one-out cross-validation, 154
level-0 models, 370–371
level-1 model, 369–371
LibLINEAR algorithm, 446t–450t, 469
LibSVM algorithm, 446t–450t, 469

lift charts, 168–172
data for, 169t
illustrated, 170f
points on, 179
lift factor, 168
linear classification
logistic regression, 125–127
using the perceptron, 127–129
using Winnow, 129–131
linear machines, 144
linear models, 62–63, 124–131
in binary classification problems, 63
boundary decision, 63
extending, 223–244
generating, 224
illustrated, 62f–63f
kernel ridge regression, 229–231
linear classification, 125–131
linear regression, 124–125
local, numeric prediction with, 251–261
logistic regression, 125–127
maximum-margin hyperplane, 224–225
in model tree, 258t
multilayer perceptrons, 232–241
nonlinear class boundaries, 226–227
numeric prediction, 124–125
perceptron, 127–129
stochastic gradient descent, 242–243
support vector machine use, 223
support vector regression, 227–229
in two dimensions, 62
linear regression, 124–125
least-squares, 63, 125–126
locally weighted, 259–261
multiple, 363
multiresponse, 125–126
linear threshold unit, 144
LinearForwardSelection method, 490t, 492–493
LinearRegression algorithm, 446t–450t, 459–462
LMT algorithm, 446t–450t, 456
load forecasting, 24–25
loading files, 416–422
locally weighted linear regression, 259–261
distance-based weighting schemes, 259–260
in nonlinear function approximation, 260
logic programs, 77
Logistic algorithm, 446t–450t, 467, 468f
logistic model trees, 368–369
logistic regression, 125–127
additive, 364–365
calibration, 346
generalizing, 126
illustrated, 127f
two-class, 126
LogitBoost, 364–365, 457, 467
LogitBoost algorithm, 475t, 476–477
log-normal distribution, 290
log-odds distribution, 290
loss functions
0 – 1, 160
informational, 161–163
quadratic, 160–163
LWL algorithm, 446t–450t, 472

M

M5′ program
CPU performance data with, 423f
error visualization, 426f
output for numeric prediction, 425f
M5P algorithm, 446t–450t, 456
M5Rules algorithm, 446t–450t, 459
machine learning, 7–8
applications, 8–9
in diagnosis applications, 25
embedded, 531–538
expert models, 352
statistics and, 28–29
machine learning schemes, 191–304
association rules, 216–223
Bayesian networks, 261–273
classification rules, 203–216
clustering, 273–293
decision trees, 192–203
instance-based learning, 244–251
linear model extensions, 223–244
multi-instance learning, 298–303
numeric prediction with local linear models, 251–261
semisupervised learning, 294–298
main() method, 553–555
MakeDensityBasedCluster algorithm, 480t, 483
MakeIndicator filter, 433t–435t, 438
makeTree() method, 540–549
manufacturing process applications, 27
market basket analysis, 26–27, 584–585
marketing and sales, 26–27
churn, 26
direct marketing, 27
historical analysis, 27
market basket analysis, 26–27
Markov blanket, 269
massive datasets, 378–380
Massive Online Analysis (MOA), 383
MathExpression filter, 433t–435t, 437, 478

maximization, 289
maximum-margin hyperplane, 224–225
  illustrated, 225f
  support vectors, 225
MDD algorithm, 446t–450t, 472–473
MDL. *See* minimum description length principle
mean-absolute errors, 181
mean-squared errors, 181
memory usage, 383
MergeTwoValues filter, 433t–435t, 438
message classifier application, 531–538
  classifyMessage() method, 537–538
  main() method, 531–536, 532f–535f
  MessageClassifier() constructor, 536
  source code, 531, 532f–535f
  updateData() method, 536–537
MetaCost algorithm, 356, 475t, 477–478
metadata, 51, 384
  application examples, 384
  extraction, 388
  knowledge, 385
  relations among attributes, 384
metalearners, 427
  configuring for boosting decision stumps, 429f
  using, 427
metalearning algorithms, in Weka, 474–479
  bagging, 474–476
  boosting, 476–477
  combining classifiers, 477
  cost-sensitive learning, 477–478
  list of, 475t
  performance optimization, 478–479
  randomization, 474–476
  retargeting classifiers, 479
methods (Weka), 520
metric trees, 137–138
MIBoost algorithm, 446t–450t, 473–474
MIDD algorithm, 446t–450t, 472–473
MIEMDD algorithm, 446t–450t, 472–473
MILR algorithm, 446t–450t, 473
minimum description length (MDL) principle, 163, 183–186
  applying to clustering, 186–187
  metric, 267
  probability theory and, 184–185
  training instances, 184
MINND algorithm, 446t–450t
MIOptimalBall algorithm, 446t–450t, 473
MISMO algorithm, 446t–450t, 473
missing values, 58–59
  1R, 87–89
  classification rules, 204–205
  decision trees, 64, 194–195
  distance function, 132
  instances with, 194
  machine learning schemes and, 58
  mixture models, 290
  Naïve Bayes, 94–97
  partial decision trees, 212
  reasons for, 58
MISVM algorithm, 446t–450t, 473
MIWrapper algorithm, 446t–450t, 473–474
mixed-attribute problems, 10–11
mixture models, 286
  extending, 289–290
  finite mixtures, 286
  missing values, 290
  nominal attributes, 289
  two-class, 286f
MOA. *See* Massive Online Analysis
model trees, 67, 251, 252
  building, 253
  illustrated, 68f
  induction pseudocode, 255–257, 256f
  linear models in, 258t
  logistic, 368–369
  with nominal attributes, 257f
  pruning, 253–254
  rules from, 259
  smoothing calculation, 252
ModelPerformanceChart, 498, 499t
MultiBoostAB algorithm, 475t, 476
multiclass prediction, 164
MultiClassClassifier algorithm, 475t, 479
multi-instance data
  classifiers, in Weka, 446t–450t, 472–474
  filters for, 440
multi-instance learning, 48, 141–143
  aggregating the input, 142
  aggregating the output, 142
  bags, 141–142, 300
  converting to single-instance learning, 298–300
  dedicated methods, 301–302
  hyperrectangles for, 303
  nearest-neighbor learning adaptation to, 301
  supervised, 141–142
  upgrading learning algorithms, 300–301
multi-instance problems, 48
  ARFF file, 55f
  converting to single-instance problem, 142
MultiInstanceToPropositional filter, 433t–435t, 440

multilabeled instances, 40
multilayer perceptrons, 232–241
  backpropagation, 235–241, 238b–239b
  bias, 233
  datasets corresponding to, 234f
  disadvantages, 238b–239b
  as feed-forward networks, 238b–239b
  hidden layer, 233, 238b–239b, 239f
  input layer, 233
  units, 233
MultilayerPerceptron algorithm, 446t–450t, 469–472
  GUI, 469, 470f
  NominalToBinaryFilter filter and, 471–472
  parameters, 471
multinominal Naïve Bayes, 97–98
multiple classes to binary transformation, 307, 338–343, 340t. *See also* data transformation
  error-correcting output codes, 339–341
  nested dichotomies, 341–343
  one-vs.-rest method, 338
  pairwise classification, 339
  pairwise coupling, 339
  simple methods, 338–339
multiple linear regression, 363
multiresponse linear regression, 125
  drawbacks, 125–126
  membership function, 125
MultiScheme algorithm, 475t, 477
multistage decision property, 103–104
multivariate decision trees, 203

N

Naïve Bayes, 93, 308
  for data streams, 381
  for document classification, 97–98
  with EM, 295
  independent attributes assumption, 289–290
  locally weighted, 260
  missing values, 94–97
  multinominal, 97–98
  numeric attributes, 94–97
  selective, 314
  semantics, 99
  visualizing, 573
  Weka algorithms, 446t–450t, 451–453
NaiveBayes algorithm, 446t–450t, 451, 452f
NaiveBayesMultinomial algorithm, 446t–450t
NaiveBayesMultinomial-Updateable algorithm, 446t–450t, 451
NaiveBayesSimple algorithm, 446t–450t, 451
NaiveBayesUpdateable algorithm, 446t–450t, 451
NAND, 233
NBTree algorithm, 446t–450t, 456
nearest-neighbor classification, 78
  speed, 137–138
nearest-neighbor learning
  attribute selection, 567–568
  class noise and, 568
  finding nearest neighbors, 88
  Hausdorff distance variants and, 303
  instance-based, 132–137
  multi-instance data adaptation, 301
  training data, varying, 569
  Weka Explorer exercise, 566–571
nested dichotomies, 341–343
  code matrix, 342t
  defined, 342
  ensemble of, 343
neural networks, 469–472
n-fold cross-validation, 154
n-grams, 387–388
Nnge algorithm, 446t–450t, 459
noise, 6–7
  class, 568
nominal attributes, 49
  mixture model, 289
  numeric prediction, 254
  symbols, 49
NominalToBinary filter, 433t–435t, 439, 444, 444t, 471–472
NominalToString filter, 433t–435t, 439
nonlinear class boundaries, 226–227
NonSparseToSparse filter, 441t, 442
normal distribution
  assumption, 97, 99
  confidence limits, 152t
normalization, 462
Normalize filter, 433t–435t, 437, 441t, 442
NOT, 233
nuclear family, 44–46
null hypothesis, 158
numeric attributes, 49, 314–322
  1R, 87–89
  classification rules, 205
  converting discrete attributes to, 322
  decision tree, 193–194
  discretization of, 306
  Naïve Bayes, 94–97
  normal-distribution assumption for, 99
numeric prediction, 15, 40
  additive regression, 362–363
  bagging for, 354–355
  evaluating, 180–182
  linear models, 124–125

numeric prediction *(continued)*
outcome as numeric value, 42
performance measures, 180t, 182t
support vector machine algorithms for, 227–228
numeric prediction (local linear models), 251–261
building trees, 253
locally weighted linear regression, 259–261
model tree induction, 255–257
model trees, 252
nominal attributes, 254
pruning trees, 253–254
rules from model trees, 259
numeric thresholds, 193
numeric-attribute problems, 10–11
NumericCleaner filter, 433t–435t, 438
NumericToBinary filter, 433t–435t, 439
NumericToNominal filter, 433t–435t, 439
NumericTransform filter, 433t–435t

O

Obfuscate filter, 433t–435t, 441
object editors, 404
generic, 417f
objects (Weka), 520
Occam's Razor, 183, 185, 361
one-class learning, 307, 335–337
multiclass classifiers, 336
outlier detection, 335–336
one-dependence estimator, 269
OneR algorithm, 446t–450t, 505
output, 458f
visualizing, 571–572
OneRAttributeEval method, 489t, 491
one-tailed probability, 151
one-vs.-rest method, 338
OPTICS algorithm, 480t, 484–485, 485f
option trees, 365–368
as alternating decision trees, 366–368, 367f
decision trees versus, 365–366
example, 366f
generation, 366
OR, 233
order-independent rules, 115
orderings, 50
circular, 51
partial, 51
ordinal attributes, 50
coding of, 51
OrdinalClassClassifier algorithm, 475t, 479
orthogonal coordinate systems, 324
outliers, 335
detection of, 335–336
output
aggregating, 142
clusters, 81
instance-based representation, 78–81
knowledge representation, 85–145
linear models, 62–63
rules, 67–77
tables, 61–62
trees, 64–67
overfitting, 88
for 1R, 88
backpropagation and, 238b–239b
forward stagewise additive regression and, 363
support vectors and, 226
overfitting-avoidance bias, 32–33
overlay data, 52

P

PaceRegression algorithm, 446t–450t, 462
packages, 519–520. *See also* Weka
weka.associations, 525
weka.attributeSelection, 525
weka.classifiers, 523–525
weka.clusterers, 525
weka.core, 520–523
weka.datagenerators, 525
weka.estimators, 525
weka.filters, 525
PageRank, 21, 375–376, 390
recomputation, 391
sink, 392
in Web mining, 391–392
pair-adjacent violators (PAV) algorithm, 345–346
paired *t*-test, 157
pairwise classification, 339
pairwise coupling, 339
parabolas, 248–249
parallelization, 379
PART algorithm, 411–413, 446t–450t, 460f–461f
partial decision trees
best leaf, 212
building example, 211f
expansion algorithm, 210f
missing values, 212
obtaining rules from, 208–212
partial least-squares regression, 326–328
partial ordering, 51
PartitionedMultiFilter filter, 433t–435t, 437–438

partitioning
for 1R, 88
discretization, 87
instance space, 80f
training set, 195
PAV. *See* pair-adjacent violators algorithm
perceptron learning rule, 128
illustrated, 129f
updating of weights, 130
perceptrons, 129
instance presentation to, 129
kernel, 231–232
linear classification using, 127–129
multilayer, 232–241
voted, 231–232
performance
classifier, predicting, 149
comparison, 147
error rate and, 148
evaluation, 148
instance-based learning, 246
for numeric prediction, 180t, 182t
optimization in Weka, 478–479
predicting, 150
text mining, 386–387
personal information use, 34–35
PKIDiscretize filter, 433t–435t, 438
PLSClassifier algorithm, 446t–450t, 462
PLSFilter filter, 444t, 445, 462
Poisson distribution, 290
postpruning, 195
subtree raising, 196–197
subtree replacement, 195–196
prediction
with Bayesian networks, 262–266
multiclass, 164
nodes, 366–367
outcomes, 164, 164t–165t
three-class, 165t
two-class, 164t
PredictionAppender, 499–500, 499t
PredictiveApriori rule learner, 486t, 487
preprocessing techniques, 574–578
prepruning, 195
principal component analysis, 324–326
of dataset, 325f
principal components, 325
recursive, 326
principal components regression, 326
PrincipalComponents filter, 433t–435t, 439
PrincipalComponents method, 489t, 491
principle of multiple explanations, 186
prior knowledge, 385
prior probability, 92–94
Prism algorithm, 446t–450t
PRISM method, 114–115, 215
probabilities
class, calibrating, 343–346
maximizing, 185
one-tailed, 151
predicting, 159–163
probability density function relationship, 96
with rules, 12–13
probability density functions, 96
probability estimates, 262
probability-based clustering, 285–286
programming by demonstration, 396
projection. *See* data projection
proportional *k*-interval discretization, 316
PropositionalToMultiInstance filter, 433t–435t, 440
pruning
cost-complexity, 202
decision trees, 195–197
example illustration, 199f
incremental reduced-error, 206, 207f
model trees, 253–254
noisy exemplars, 245–246
postpruning, 195
prepruning, 195
reduced-error, 197, 206
rules, 200–201
subtree lifting, 199–200
subtree raising, 196–197
subtree replacement, 195–196
pruning sets, 205–206

Q

quadratic loss function, 160–163

R

RacedIncrementalLogitBoost algorithm, 475t, 476–477
race search, 313
RaceSearch method, 490t, 493
radial basis function (RBF), 241–242
kernels, 227
networks, 227
output layer, 241–242
random projections, 326
random subspaces, 357
RandomCommittee algorithm, 475t, 476
RandomForest algorithm, 446t–450t

randomization, 356–358
bagging versus, 357
results, 356–357
rotation forests, 357–358
in Weka, 474–476
Randomize filter, 441t, 442
randomizing
unsupervised attribute filters, 441
unsupervised instance filters, 442
RandomProjection filter, 433t–435t, 441
RandomSearch method, 490t, 493
RandomSubset filter, 433t–435t, 437, 476
RandomSubSpace algorithm, 475t
RandomTree algorithm, 446t–450t, 455
Ranker method, 490–491, 490t, 494
RankSearch method, 490t, 493–494
ratio quantities, 50
RBF. *See* radial basis function
RBFNetwork algorithm, 446t–450t, 467–469
recall-precision curves, 174–175
AUPRC, 177
points on, 179
rectangular generalizations, 79
recurrent neural networks, 238b–239b
recursive feature elimination, 309–310
reduced-error pruning, 206, 238
incremental, 206, 207f
reference density, 337
reference distribution, 337
regression, 15, 62
additive, 362–365
isotonic, 345
kernel ridge, 229–231
linear, 124–125
locally weighted, 259–261
logistic, 125–127
partial least-squares, 326–328
principal components, 326
robust, 333–334
support vector, 227–229
regression equations, 15
regression tables, 61–62
regression trees, 67, 251
illustrated, 68f
RegressionByDiscretization algorithm, 475t, 479
regularization, 244
reidentification, 33–34
RELAGGS filter, 433t–435t, 440
RELAGGS system, 302–303
relations, 43–46
ancestor-of, 46
sister-of, 43f, 45t
superrelations, 44–46
relation-valued attributes, 54–55
instances, 56–57
specification, 55
relative-absolute errors, 181
relative-squared errors, 181
RELIEF (Recursive Elimination of Features), 346
ReliefFAttributeEval method, 489t, 490–491
reloading datasets, 418–419
Remove filter, 433t–435t, 436
RemoveFolds filter, 441t, 442
RemoveFrequentValues filter, 441t, 442
RemoveMisclassified filter, 441t, 442
RemovePercentage filter, 441t, 442
RemoveRange filter, 441t, 442
RemoveType filter, 433t–435t, 436
RemoveUseless filter, 433t–435t, 436
RemoveWithValues filter, 441t, 442
Reorder filter, 433t–435t, 437–438
repeated holdout, 152–153
ReplaceMissingValues filter, 433t–435t, 438
replicated subtree problem, 69
decision tree illustration, 71f
REPTree algorithm, 446t–450t, 456
Resample filter, 441t, 442, 444t, 445
reservoir sampling, 330–331
ReservoirSample filter, 441t, 442
residuals, 327, 368–369
resubstitution errors, 148–149
retargeting classifiers, in Weka, 479
Ridor algorithm, 446t–450t, 459
RIPPER algorithm, 208, 209f, 215
ripple-down rules, 216
robo-soccer, 394
robust regression, 333–334
ROC curves, 172–174, 581
AUC, 177
from different learning schemes, 173–174
generating with cross-validation, 173
jagged, 172–173
points on, 179
sample, 173f
for two learning schemes, 174f
rotation forests, 357–358
RotationForest algorithm, 475t, 476
rote learning, 78
row separation, 340
rule sets
model trees for generating, 259
for noisy data, 203
visualizing, 573
rules, 10, 67–77
antecedent of, 67
association, 11, 72–73, 216–223

classification, 11, 69–72
computer-generated, 19–21
consequent of, 67
constructing, 108–116
decision lists versus, 115–116
decision tree, 200–201
efficient generation of, 122–123
with exceptions, 73–75, 212–215
expert-derived, 19–21
expressive, 75–77
inferring, 86–90
from model trees, 259
order-independent, 115
perceptron learning, 128
popularity, 70–71
PRISM method for constructing, 114–115
probabilities, 12–13
pruning, 200–201
ripple-down, 216
trees versus, 109–110
Weka algorithms, 446t–450t, 457–459

S

sampling, 307, 330–331. *See also* data transformation
with replacement, 330–331
reservoir, 330–331
without replacement, 330
ScatterPlotMatrix, 498, 499t
ScatterSearchV1 method, 490t, 494
schemata search, 313
scheme-independent attribute selection, 308–310
filter method, 308–309
instance-based learning methods, 310
recursive feature elimination, 309–310
symmetric uncertainty, 310b
wrapper method, 308–309
scheme-specific attribute selection, 312–314
accelerating, 313
paired *t*-test, 313
race search, 313
results, 312–313
schemata search, 313
selective Naïve Bayes, 314
scheme-specific options, 528t, 529
scientific applications, 28
screening images, 23–24
SDR. *See* standard deviation reduction
search, generalization as, 29
search bias, 32
search engines, in web mining, 21–22
search methods (Weka), 421, 490t
seeds, 139
selective Naïve Bayes, 314
semantic relationship, 384
semisupervised learning, 294–298
clustering for classification, 294–296
co-EM, 297
co-training, 296
separate-and-conquer algorithms, 115–116, 308
SerializedClassifier algorithm, 474
SerializedModelSaver, 499–500, 499t
set kernel, 301
shapes problem, 75
illustrated, 76f
training data, 76t
sIB algorithm, 480t, 485
sigmoid function, 236f
sigmoid kernel, 227
SimpleCart algorithm, 446t–450t, 456
SimpleKMeans algorithm, 480–481, 480t, 481f
SimpleLinearRegression algorithm, 446t–450t, 459, 461f
SimpleLogistic algorithm, 446t–450t, 467
SimpleMI algorithm, 446t–450t, 473–474
single-attribute evaluators, 490–492
single-consequent rules, 123
single-linkage clustering algorithm, 275, 279
skewed datasets, 135
SMO algorithm, 446t–450t, 462, 463f–467f
smoothing calculation, 252
SMOreg algorithm, 446t–450t, 462
SMOTE filter, 444t, 445
soybean classification example, 5
dataset, 20t
examples rules, 19
sparse data, 56
sparse instances, 442
SparseToNonSparse filter, 441t, 442
SPegasos algorithm, 446t–450t, 464
splitter nodes, 366–367
splitting, 281
clusters, 274
criterion, 253
model tree nodes, 255
SpreadSubsample filter, 444t, 445
SQLViewer, 419
squared error, 161
stacking, 334, 369–371
defined, 144, 369
level-0 model, 370–371
level-1 model, 369–371
model input, 369
output combination, 369
as parallel, 379

Stacking algorithm, 475t, 477
StackingC algorithm, 475t, 477
standard deviation from the mean, 151
standard deviation reduction (SDR), 253, 254
Standardize filter, 433t–435t, 437
standardizing statistical variables, 57
statistical clustering, 314–315
statistical modeling, 90–99
statistics, machine learning and, 28–29
step function, 236f
stochastic backpropagation, 238b–239b
stochastic gradient descent, 242–243
stopwords, 329, 387
stratification, 152
  variation reduction, 153–154
stratified holdout, 152
stratified threefold cross-validation, 153
StratifiedRemoveFolds filter, 444t, 445
StreamableFilter keyword, 526
string attributes, 54
  in document classification, 579–580
  specification, 54
  values, 54
StringToNominal filter, 433t–435t, 439
StringToWordVector filter, 419, 433t–435t, 439–440, 538
  default, 581
  options, 581
StripChart, 498, 499t
structural descriptions, 5–7
  decision trees, 5
  learning techniques, 8–9
structure learning by conditional independence tests, 270
Student's distribution with k-1 degrees of freedom, 157
Student's t-test, 157
subgradients, 242
subsampling, 442
SubsetByExpression filter, 441t, 442
SubsetSizeForwardSelection method, 490t, 492–493
subtree lifting, 199–200
subtree raising, 196–197
subtree replacement, 195–196
success rate, error rate and, 197–198
superparent one-dependence estimator, 269
superrelations, 44–46
supervised discretization, 316, 574
supervised filters, 432, 443–445
  attribute, 443–445
  instance, 445
  using, 432
supervised learning, 40
support, of association rules, 72, 116
support vector machines (SVMs), 191–192
  co-EM with, 297
  hinge loss, 242–243
  linear model usage, 223
  term usage, 223
  training, 225
  weight update, 243
support vector regression, 227–229
  flatness maximization, 229
  illustrated, 228f
  for linear case, 229
  linear regression differences, 228
  for nonlinear case, 229
support vectors, 191–192, 225
  finding, 225
  overfitting and, 226
SVMAttributeEval method, 489t, 491
SwapValues filter, 433t–435t, 438
symmetric uncertainty, 310b
SymmetricalUncertAttributeEval method, 489t, 491

T

tables
  as knowledge representation, 61–62
  regression, 61–62
tabular input format, 124
TAN. *See* tree-augmented Naïve Bayes
teleportation, 392
tenfold cross-validation, 153–154, 306
Tertius rule learner, 486t, 487
testing, 148–150
  test data, 149
  test sets, 149
  in Weka, 422–424
TestSetMaker, 499–502, 499t
text mining, 386–389
  data mining versus, 386–387
  document classification, 387–388
  entity extraction, 388
  information extraction, 388–389
  metadata extraction, 388
  performance, 386–387
  stopwords, 387
text summarization, 387
text to attribute vectors, 328–329
TextViewer, 499t

theory, 183
exceptions to, 183
MDL principle and, 183–184
threefold cross-validation, 153
three-point average recall, 175
ThresholdSelector algorithm, 475t, 479
time series, 330
Delta, 330
filters for, 440
timestamp attribute, 330
TimeSeriesDelta filter, 433t–435t, 440
TimeSeriesTranslate filter, 433t–435t, 440
timestamp attribute, 330
tokenization, 328–329, 440
top-down induction, of decision trees, 107–108
toSource() method, 550–553
training, 148–150
data, 149
data verification, 569
documents, 579t
instances, 184
learning schemes (Weka), 422–424
support vector machines, 225
training sets, 147
error, 197
error rate, 148
partitioning, 195
size effects, 569t
TrainingSetMaker, 499–502, 499t
TrainTestSplitMaker, 499–500, 499t
tree diagrams. *See* dendrograms
tree-augmented Naïve Bayes (TAN), 269
trees, 64–67. *See also* decision trees
AD, 270–272, 271f
ball, 135–137
frequent-pattern, 216–219
functional, 65
Hoeffding, 382–383
kD, 132–133, 133f–134f
logistic model, 368–369
metric, 137–138
model, 67, 68f, 251–252
option, 365–368
regression, 67, 68f, 251
rules versus, 109–110
Weka algorithms, 416, 446t–450t
trees package, 519–520
true negatives (TN), 164, 580
true positive rate, 164
true positives (TP), 164, 580
t-statistic, 158–159
t-test, 157
corrected resampled, 159
paired, 157
two-class mixture model, 286f
two-class problem, 75
typographic errors, 59

U

ubiquitous computing, 395–396
ubiquitous data mining, 395–397
univariate decision trees, 203
unmasking, 394–395
unsupervised attribute filters, 432–441. *See also* filtering algorithms; filters
adding/removing attributes, 436–438
changing values, 438
conversions, 438–439
list of, 433t–435t
multi-instance data, 440
randomizing, 441
string conversion, 439–440
time series, 440
unsupervised discretization, 316, 574
unsupervised instance filters, 441–442
list of, 441t
randomizing, 442
sparse instances, 442
subsampling, 442
UpdateableClassifier keyword, 526
updateData() method, 536–537
User Classifier (Weka), 65, 424–427
segmentation data with, 428f
UserClassifier algorithm, 446t–450t, 570

V

validation data, 149
validation sets, 379
variables, standardizing, 57
variance, 354
Venn diagrams, in cluster representation, 81
VFI algorithm, 417, 446t–450t
visualization
Bayesian network, 454f
classification errors, 565
decision trees, 573
Naïve Bayes, 573
nearest-neighbor learning, 572
OneR, 571–572
rule sets, 573
in Weka, 430–432

Visualize panel, 430–432, 562
Vote algorithm, 475t, 477
voted perceptron, 197
VotedPerceptron algorithm, 446t–450t, 464
voting feature intervals, defined, 138

W

WAODE algorithm, 446t–450t, 451
Wavelet filter, 433t–435t, 439
weather problem example, 9–12
  alternating decision tree, 367f
  ARFF file for, 53f, 409f
  association rules, 11, 120t–121t
  attribute space, 311f
  attributes, 9–10
  attributes evaluation, 87t
  Bayesian network visualization, 454f
  Bayesian networks, 263f, 265f
  clustering, 280f
  counts and probabilities, 91t
  CSV format for, 409f
  data with numeric class, 42t
  dataset, 10t
  decision tree, 103f
  EM output, 482f
  expanded tree stumps, 102f
  FP-tree insertion, 217t–218t
  identification codes, 106t
  item sets, 117t–118t
  multi-instance ARFF file, 55f
  NaiveBayes output, 452f
  numeric data with summary statistics, 95t
  option tree, 366f
  SimpleKMeans output, 481f
  tree stumps, 100f
web mining, 5, 389–392
  PageRank algorithm, 390–392
  search engines, 21–22
  teleportation, 392
  wrapper induction, 390
weight decay, 238b–239b
weighting attributes
  instance-based learning, 246–247
  test, 247
  updating, 247
weights
  determination process, 15
  with rules, 12–13
Weka, 403–406
  advanced setup, 511–512
  ARFF format, 407
  association rule mining, 582–584
  association rules, 429–430
  association-rule learners, 485–487
  attribute selection, 430, 487–494
  clustering, 429
  clustering algorithms, 480–485
  command-line interface, 519–530
  components configuration and connection, 500–502
  CPU performance data, 423f
  data preparation, 407
  development of, 403
  evaluation components, 498–500, 499t
  experiment distribution, 515–517
  Experimenter, 405, 505–517
  Explorer, 404, 407–494
  filtering algorithms, 432–445
  Generic Object Editor, 417f
  GUI Chooser panel, 408
  how to use, 404–405
  incremental learning, 502–503
  interfaces, 404–405
  ISO-8601 date/time format, 54
  Knowledge Flow, 404–405, 495–503
  learning algorithms, 445–474
  learning scheme creation, 539–557
  market basket analysis, 584–585
  message classifier application, 531–538
  metalearning algorithms, 474–479
  neural networks, 469–472
  packages, 519–525
  search methods, 492–494
  simple setup, 510–511
  structure of, 519–526
  User Classifier facility, 65, 424–427
  visualization, 430–432
  visualization components, 498–500, 499t
weka.associations package, 525
weka.attributeSelection package, 525
weka.classifiers package, 523–525
  DecisionStump class, 523, 524f
  implementations, 523
weka.classifiers.trees.Id3, 539–555
  buildClassifier() method, 540
  classifyInstance() method, 549–550
  computeInfoGain() method, 549
  getCapabilities() method, 539
  getTechnicalInformation() method, 539
  globalInfo() method, 539
  main() method, 553–555
  makeTree() method, 540–549
  Sourcable interface, 539, 550
  source code, 541f–548f
  source code for weather example, 551f–553f

TechnicalInformationHandler interface, 539
toSource() method, 550–553
weka.clusterers package, 525
weka.core package, 520–523
classes, 523
web page illustration, 521f–522f
weka.datagenerators package, 525
weka.estimators package, 525
weka.filters package, 525
weka.log, 415–416
weka package, 520
Weka workbench, 376, 403
filters, 404
J4.8 algorithm, 410–414
Winnow, 129–130
Balanced, 131
linear classification with, 88
updating of weights, 130
versions illustration, 130f
Winnow algorithm, 446t–450t
wisdom, 35
wrapper induction, 390
wrapper method, 308–309
wrappers, 389–390
WrapperSubsetEval method, 488, 489t

X

XMeans algorithm, 480t, 483
XML (eXtensible Markup Language), 52–56
XOR (exclusive-OR), 233
XRFF format, 419

Z

zero-frequency problem, 162
ZeroR algorithm, 413, 446t–450t, 459, 505