

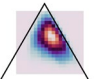
Mi clasificador es mejor que el tuyo

Breve introducción a la comparación estadística de clasificadores

Por: Mario Juez-Gil

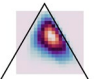


Qué hago aquí



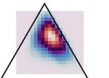
Qué hago aquí

- Investigo sobre *Machine Learning* y quiero saber cómo comparar clasificadores.



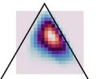
Qué hago aquí

- Investigo sobre *Machine Learning* y quiero saber cómo comparar clasificadores.
- Quiero aprender a interpretar los análisis comparativos que he visto en algunos *papers*.



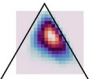
Qué hago aquí

- Investigo sobre *Machine Learning* y quiero saber cómo comparar clasificadores.
- Quiero aprender a interpretar los análisis comparativos que he visto en algunos *papers*.
- Josema y José Antonio me han engañado para que venga.

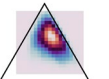


Hoja de ruta

- Contexto: Machine learning, entrenamiento, validación cruzada, métricas de rendimiento...
- Comparación estadística basada en contraste de hipótesis
- Comparación bayesiana de clasificadores
- Líneas de trabajo futuras en este ámbito



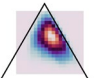
Machine Learning



Machine Learning

Algoritmo

```
163  /**
164   * Trains a Rotation Forest classification model.
165   *
166   * @param dataset Training dataset.
167   * @return Rotation Forest classification model.
168   */
169  @Since("2.4.4")
170  protected def train(dataset: Dataset[_]): RotationForestClassificationModel = {
171    val categoricalFeatures =
172      MetadataUtils.getCategoricalFeatures(dataset.schema($(featuresCol)))
173    require(categoricalFeatures.size == 0, "Rotation Forest cannot work with " +
174      "categorical features. Consider transforming the categorical features " +
175      "using OneHotEncoderEstimator for example.")
176
177    val numClasses = MetadataUtils.getNumClasses(dataset.schema($(labelCol))) match {
178      case Some(n: Int) => n
179      case None => throw new IllegalArgumentException("The dataset must " +
180        "contain a label column with spark ML metadata. See spark ml " +
181        "StringIndexer transformer for more information.")
182    }
183
184    val rnd = new Random($(seed))
185
186    val numFeatures = countFeatures(dataset)
187    val classes = 0d to (numClasses - 1) by 1d toList
188
189    val (scaledDataset, scaler) = scaleDatasetAndGetScalerModel(dataset)
190    val cachedDataset = scaledDataset.cache()
191    val ensemble = buildEnsemble(cachedDataset, numFeatures, classes, rnd)
192    cachedDataset.unpersist()
193
194    new RotationForestClassificationModel(ensemble, scaler, numClasses)
195  }
```



Machine Learning

Datos

Algoritmo

Entrenamiento

```
163 /**
164  * Trains a Rotation Forest classification model.
165  *
166  * @param dataset Training dataset.
167  * @return Rotation Forest classification model.
168  */
169 @Since("2.4.4")
170 protected def train(dataset: Dataset[_]): RotationForestClassificationModel = {
171   val categoricalFeatures =
172     MetadataUtils.getCategoricalFeatures(dataset.schema($(featuresCol)))
173   require(categoricalFeatures.size == 0, "Rotation Forest cannot work with " +
174     "categorical features. Consider transforming the categorical features " +
175     "using OneHotEncoderEstimator for example.")
176
177   val numClasses = MetadataUtils.getNumClasses(dataset.schema($(labelCol))) match {
178     case Some(n: Int) => n
179     case None => throw new IllegalArgumentException("The dataset must " +
180       "contain a label column with spark ML metadata. See spark ml " +
181       "StringIndexer transformer for more information.")
182   }
183
184   val rnd = new Random($(seed))
185
186   val numFeatures = countFeatures(dataset)
187   val classes = 0d to (numClasses - 1) by 1d toList
188
189   val (scaledDataset, scaler) = scaleDatasetAndGetScalerModel(dataset)
190   val cachedDataset = scaledDataset.cache()
191   val ensemble = buildEnsemble(cachedDataset, numFeatures, classes, rnd)
192   cachedDataset.unpersist()
193
194   new RotationForestClassificationModel(ensemble, scaler, numClasses)
195 }
```

0	128:51	129:159	130:253	131:159	132:50	155:48	156:238	157:252	158:252	159:252	160:237	182:54	18
1	159:124	160:253	161:255	162:63	186:96	187:244	188:251	189:253	190:62	214:127	215:251	216:251	2
1	125:145	126:255	127:211	128:31	152:32	153:237	154:253	155:252	156:71	180			
1	153:5	154:63	155:197	181:20	182:254	183:230	184:24	209:20	210:254	211:25			
1	152:1	153:168	154:242	155:28	180:10	181:228	182:254	183:100	209:190	210:			
0	130:64	131:253	132:255	133:63	157:96	158:205	159:251	160:253	161:205	162			
1	159:121	160:254	161:136	186:13	187:230	188:253	189:248	190:99	213:4	214:			
1	100:166	101:222	102:55	128:197	129:254	130:218	131:5	155:29	156:249	157:254	158:254	159:	183:
0	155:53	156:255	157:253	158:253	159:253	160:124	183:180	184:253	185:251	186:251	187:5	188:	251
0	128:73	129:253	130:227	131:73	132:21	156:73	157:251	158:251	159:251	160:174	181:	183:	166
1	155:178	156:255	157:105	182:6	183:188	184:253	185:216	186:14	210:14	211:	212:	253	21
0	154:46	155:105	156:254	157:254	158:254	159:254	160:255	161:	162:41	180:37	181:118	182:	222
0	152:56	153:105	154:220	155:254	156:63	178:	180:	180:233	181:253	182:253	183:253	184:	236
1	130:7	131:176	132:	133:	134:	135:	136:	137:	138:	139:	140:	141:	142
0	155:21	156:176	157:253										
1	151:68	152:45	153:131										
0	125:29	126:170	127:255										
0	153:203	154:254	155:25										
1	98:64	99:191	100:70	125:68	126:243	127:253	128:249	129:63	152:30	153:223	154:253	155:253	156:2
1	125:26	126:240	127:72	153:25	154:238	155:208	182:209	183:226	184:14	210:209	211:254	212:43	238
0	155:62	156:91	157:213	158:255	159:228	160:91	161:12	182:70	183:230	184:253	185:253	186:	253
1	157:42	158:228	159:253	160:253	185:144	186:251	187:251	188:251	212:89	213:236	214:251	215:	235
1	128:62	129:254	130:213	156:102	157:253	158:252	159:102	160:20	184:102	185:254	186:253	187:	254
0	154:28	155:195	156:254	157:254	158:254	159:254	160:254	161:255	162:61	181:6	182:191	183:	253
0	123:8	124:76	125:202	126:254	127:255	128:163	129:37	130:2	150:13	151:182	152:253	153:	253



Machine Learning

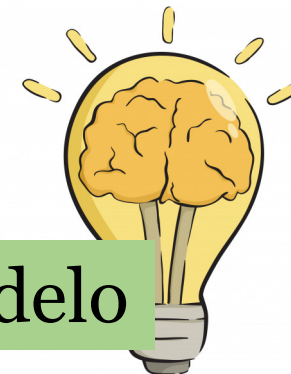
```
0 128:51 129:159 130:253 131:159 132:50 155:48 156:238 157:252 158:252 159:252 160:237 182:54 18
1 159:124 160:253 161:255 162:63 186:96 187:244 188:251 189:253 190:62 214:127 215:251 216:251 2
1 125:145 126:255 127:211 128:31 152:32 153:237 154:253 155:252 156:71 180
1 153:5 154:63 155:197 181:20 182:254 183:230 184:24 209:20 210:254 211:25
1 152:1 153:168 154:242 155:28 180:10 181:228 182:254 183:100 209:190 210:
0 130:64 131:253 132:255 133:63 157:96 158:205 159:251 160:253 161:205 162
1 159:121 160:254 161:136 186:13 187:230 188:253 189:248 190:99 213:4 214:
1 100:166 101:222 102:55 128:197 129:254 130:218 131:5 155:29 156:249 157:254 158:254 159:
0 155:53 156:255 157:253 158:253 159:253 160:124 183:180 184:253 185:251 186:251 187:
0 128:73 129:253 130:227 131:73 132:21 156:73 157:251 158:251 159:251 160:174 183:
1 155:178 156:255 157:105 182:6 183:188 184:253 185:216 186:14 210:14 211:
0 154:46 155:105 156:254 157:254 158:254 159:254 160:255 161:
0 152:56 153:105 154:220 155:254 156:63 178:
1 130:7 131:176 132:
0 155:21 156:176 157:253
1 151:68 152:45 153:131
0 125:29 126:170 127:255
0 153:203 154:254 155:25
1 98:64 99:191 100:70 125:68 126:243 127:253 128:249 129:63 152:30 153:223 154:253 155:253 156:2
1 125:26 126:240 127:72 153:25 154:238 155:208 182:209 183:226 184:14 210:209 211:254 212:43 238
0 155:62 156:91 157:213 158:255 159:228 160:91 161:12 182:70 183:230 184:253 185:253 186:253 187
1 157:42 158:228 159:253 160:253 185:144 186:251 187:251 188:251 212:89 213:236 214:251 215:235
1 128:62 129:254 130:213 156:102 157:253 158:252 159:102 160:20 184:102 185:254 186:253 187:254
0 154:28 155:195 156:254 157:254 158:254 159:254 160:254 161:255 162:61 181:6 182:191 183:253 18
0 123:8 124:76 125:202 126:254 127:255 128:163 129:37 130:2 150:13 151:182 152:253 153:253 154:2
```

Datos

Algoritmo

Entrenamiento

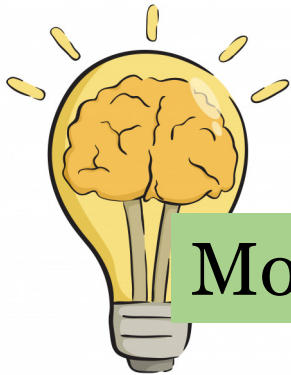
Descubrimiento de patrones



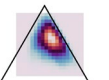
Modelo

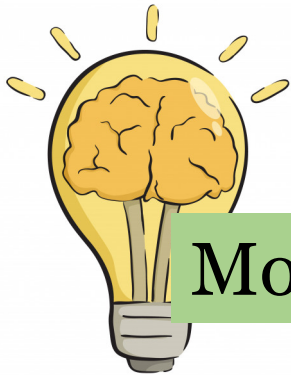
```
163 /**
164  * Trains a Rotation Forest classification model.
165  *
166  * @param dataset Training dataset.
167  * @return Rotation Forest classification model.
168  */
169 @Since("2.4.4")
170 protected def train(dataset: Dataset[_]): RotationForestClassificationModel =
171   val categoricalFeatures =
172     MetadataUtils.getCategoricalFeatures(dataset.schema($(featuresCol)))
173   require(categoricalFeatures.size == 0, "Rotation Forest cannot work with " +
174     "categorical features. Consider transforming the categorical features " +
175     "using OneHotEncoderEstimator for example.")
176
177   val numClasses = MetadataUtils.getNumClasses(dataset.schema($(labelCol))) match {
178     case Some(n: Int) => n
179     case None => throw new IllegalArgumentException("The dataset must " +
180       "contain a label column with spark ML metadata. See spark ml " +
181       "StringIndexer transformer for more information.")
182   }
183
184   val rnd = new Random($(seed))
185
186   val numFeatures = countFeatures(dataset)
187   val classes = 0d to (numClasses - 1) by 1d toList
188
189   val (scaledDataset, scaler) = scaleDatasetAndGetScalerModel(dataset)
190   val cachedDataset = scaledDataset.cache()
191   val ensemble = buildEnsemble(cachedDataset, numFeatures, classes, rnd)
192   cachedDataset.unpersist()
193
194   new RotationForestClassificationModel(ensemble, scaler, numClasses)
195 }
```





Modelo



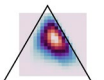


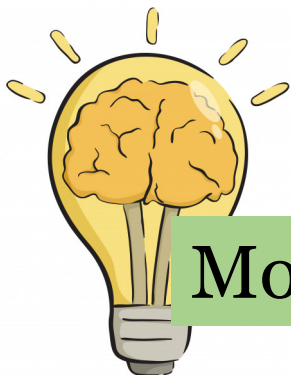
Modelo

```
0 128:51 129:159 130:253 131:159 132:50 155:40 156:238 157:252 158:252 159:252 160:237 182:54 18
1 159:124 160:253 161:255 162:63 186:96 187:244 188:251 189:253 190:62 214:127 215:251 216:251 2
1 125:145 126:255 127:211 128:31 152:32 153:237 154:253 155:252 156:71 188
1 153:5 154:63 155:197 181:20 182:254 183:230 184:24 209:20 210:254 211:25
1 152:1 153:168 154:242 155:28 180:10 181:228 182:254 183:100 209:190 210:
0 130:64 131:253 132:255 133:63 157:96 158:205 159:251 160:253 161:205 162
1 159:121 160:254 161:136 186:13 187:230 188:253 189:248 190:99 213:4 214:
1 100:166 101:222 102:55 128:197 129:254 130:218 131:5 155:29 156:249 157:254 158:254 159:
0 155:53 156:255 157:253 158:253 159:253 160:124 183:180 184:253 185:251 186:251 187:
0 128:73 129:253 130:227 131:73 132:21 156:73 157:251 158:251 159:251 160:174
1 155:178 156:255 157:185 182:6 183:188 184:253 185:216 186:14 210:1
0 154:46 155:105 156:254 157:254 158:254 159:254 160:254 161:254 162:41 180:37 181:118 182:222 1
0 152:56 153:105 154:220 155:254 156:254 157:254 179:166 180:233 181:253 182:253 183:253 184:236 1
0 152:77 153:224 133:224 158:51 159:253 160:253 161:233 185:4 186:170 187:253 188:253 189
0 155:21 156:176 157:253 158:253 159:253 160:253 161:253 162:253 163:253 164:253 165:253 166:253 167:253 168:253 169:253 170:253 171:253 172:253 173:253 174:253 175:253 176:253 177:253 178:253 179:253 180:253 181:253 182:253 183:253 184:253 185:253 186:253 187:253 188:253 189:253 190:253 191:253 192:253 193:253 194:253 195:253 196:253 197:253 198:253 199:253 200:253 201:253 202:253 203:253 204:253 205:253 206:253 207:253 208:253 209:253 210:253 211:253 212:253 213:253 214:253 215:253 216:253 217:253 218:253 219:253 220:253 221:253 222:253 223:253 224:253 225:253 226:253 227:253 228:253 229:253 230:253 231:253 232:253 233:253 234:253 235:253 236:253 237:253 238:253 239:253 240:253 241:253 242:253 243:253 244:253 245:253 246:253 247:253 248:253 249:253 250:253 251:253 252:253 253:253 254:253 255:253 256:253 257:253 258:253 259:253 260:253 261:253 262:253 263:253 264:253 265:253 266:253 267:253 268:253 269:253 270:253 271:253 272:253 273:253 274:253 275:253 276:253 277:253 278:253 279:253 280:253 281:253 282:253 283:253 284:253 285:253 286:253 287:253 288:253 289:253 290:253 291:253 292:253 293:253 294:253 295:253 296:253 297:253 298:253 299:253 300:253 301:253 302:253 303:253 304:253 305:253 306:253 307:253 308:253 309:253 310:253 311:253 312:253 313:253 314:253 315:253 316:253 317:253 318:253 319:253 320:253 321:253 322:253 323:253 324:253 325:253 326:253 327:253 328:253 329:253 330:253 331:253 332:253 333:253 334:253 335:253 336:253 337:253 338:253 339:253 340:253 341:253 342:253 343:253 344:253 345:253 346:253 347:253 348:253 349:253 350:253 351:253 352:253 353:253 354:253 355:253 356:253 357:253 358:253 359:253 360:253 361:253 362:253 363:253 364:253 365:253 366:253 367:253 368:253 369:253 370:253 371:253 372:253 373:253 374:253 375:253 376:253 377:253 378:253 379:253 380:253 381:253 382:253 383:253 384:253 385:253 386:253 387:253 388:253 389:253 390:253 391:253 392:253 393:253 394:253 395:253 396:253 397:253 398:253 399:253 400:253 401:253 402:253 403:253 404:253 405:253 406:253 407:253 408:253 409:253 410:253 411:253 412:253 413:253 414:253 415:253 416:253 417:253 418:253 419:253 420:253 421:253 422:253 423:253 424:253 425:253 426:253 427:253 428:253 429:253 430:253 431:253 432:253 433:253 434:253 435:253 436:253 437:253 438:253 439:253 440:253 441:253 442:253 443:253 444:253 445:253 446:253 447:253 448:253 449:253 450:253 451:253 452:253 453:253 454:253 455:253 456:253 457:253 458:253 459:253 460:253 461:253 462:253 463:253 464:253 465:253 466:253 467:253 468:253 469:253 470:253 471:253 472:253 473:253 474:253 475:253 476:253 477:253 478:253 479:253 480:253 481:253 482:253 483:253 484:253 485:253 486:253 487:253 488:253 489:253 490:253 491:253 492:253 493:253 494:253 495:253 496:253 497:253 498:253 499:253 500:253 501:253 502:253 503:253 504:253 505:253 506:253 507:253 508:253 509:253 510:253 511:253 512:253 513:253 514:253 515:253 516:253 517:253 518:253 519:253 520:253 521:253 522:253 523:253 524:253 525:253 526:253 527:253 528:253 529:253 530:253 531:253 532:253 533:253 534:253 535:253 536:253 537:253 538:253 539:253 540:253 541:253 542:253 543:253 544:253 545:253 546:253 547:253 548:253 549:253 550:253 551:253 552:253 553:253 554:253 555:253 556:253 557:253 558:253 559:253 560:253 561:253 562:253 563:253 564:253 565:253 566:253 567:253 568:253 569:253 570:253 571:253 572:253 573:253 574:253 575:253 576:253 577:253 578:253 579:253 580:253 581:253 582:253 583:253 584:253 585:253 586:253 587:253 588:253 589:253 590:253 591:253 592:253 593:253 594:253 595:253 596:253 597:253 598:253 599:253 600:253 601:253 602:253 603:253 604:253 605:253 606:253 607:253 608:253 609:253 610:253 611:253 612:253 613:253 614:253 615:253 616:253 617:253 618:253 619:253 620:253 621:253 622:253 623:253 624:253 625:253 626:253 627:253 628:253 629:253 630:253 631:253 632:253 633:253 634:253 635:253 636:253 637:253 638:253 639:253 640:253 641:253 642:253 643:253 644:253 645:253 646:253 647:253 648:253 649:253 650:253 651:253 652:253 653:253 654:253 655:253 656:253 657:253 658:253 659:253 660:253 661:253 662:253 663:253 664:253 665:253 666:253 667:253 668:253 669:253 670:253 671:253 672:253 673:253 674:253 675:253 676:253 677:253 678:253 679:253 680:253 681:253 682:253 683:253 684:253 685:253 686:253 687:253 688:253 689:253 690:253 691:253 692:253 693:253 694:253 695:253 696:253 697:253 698:253 699:253 700:253 701:253 702:253 703:253 704:253 705:253 706:253 707:253 708:253 709:253 710:253 711:253 712:253 713:253 714:253 715:253 716:253 717:253 718:253 719:253 720:253 721:253 722:253 723:253 724:253 725:253 726:253 727:253 728:253 729:253 730:253 731:253 732:253 733:253 734:253 735:253 736:253 737:253 738:253 739:253 740:253 741:253 742:253 743:253 744:253 745:253 746:253 747:253 748:253 749:253 750:253 751:253 752:253 753:253 754:253 755:253 756:253 757:253 758:253 759:253 760:253 761:253 762:253 763:253 764:253 765:253 766:253 767:253 768:253 769:253 770:253 771:253 772:253 773:253 774:253 775:253 776:253 777:253 778:253 779:253 780:253 781:253 782:253 783:253 784:253 785:253 786:253 787:253 788:253 789:253 790:253 791:253 792:253 793:253 794:253 795:253 796:253 797:253 798:253 799:253 800:253 801:253 802:253 803:253 804:253 805:253 806:253 807:253 808:253 809:253 810:253 811:253 812:253 813:253 814:253 815:253 816:253 817:253 818:253 819:253 820:253 821:253 822:253 823:253 824:253 825:253 826:253 827:253 828:253 829:253 830:253 831:253 832:253 833:253 834:253 835:253 836:253 837:253 838:253 839:253 840:253 841:253 842:253 843:253 844:253 845:253 846:253 847:253 848:253 849:253 850:253 851:253 852:253 853:253 854:253 855:253 856:253 857:253 858:253 859:253 860:253 861:253 862:253 863:253 864:253 865:253 866:253 867:253 868:253 869:253 870:253 871:253 872:253 873:253 874:253 875:253 876:253 877:253 878:253 879:253 880:253 881:253 882:253 883:253 884:253 885:253 886:253 887:253 888:253 889:253 890:253 891:253 892:253 893:253 894:253 895:253 896:253 897:253 898:253 899:253 900:253 901:253 902:253 903:253 904:253 905:253 906:253 907:253 908:253 909:253 910:253 911:253 912:253 913:253 914:253 915:253 916:253 917:253 918:253 919:253 920:253 921:253 922:253 923:253 924:253 925:253 926:253 927:253 928:253 929:253 930:253 931:253 932:253 933:253 934:253 935:253 936:253 937:253 938:253 939:253 940:253 941:253 942:253 943:253 944:253 945:253 946:253 947:253 948:253 949:253 950:253 951:253 952:253 953:253 954:253 955:253 956:253 957:253 958:253 959:253 960:253 961:253 962:253 963:253 964:253 965:253 966:253 967:253 968:253 969:253 970:253 971:253 972:253 973:253 974:253 975:253 976:253 977:253 978:253 979:253 980:253 981:253 982:253 983:253 984:253 985:253 986:253 987:253 988:253 989:253 990:253 991:253 992:253 993:253 994:253 995:253 996:253 997:253 998:253 999:253 1000:253
```

Datos

Predicción





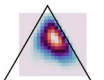
Modelo

```
0 128:51 129:159 130:253 131:159 132:50 155:40 156:238 157:252 158:252 159:252 160:237 182:54 18
1 159:124 160:253 161:255 162:63 186:96 187:244 188:251 189:253 190:62 214:127 215:251 216:251 2
1 125:145 126:255 127:211 128:31 152:32 153:237 154:253 155:252 156:71 188
1 153:5 154:63 155:197 181:20 182:254 183:230 184:24 209:20 210:254 211:25
1 152:1 153:168 154:242 155:28 180:10 181:228 182:254 183:100 209:190 210:
0 130:64 131:253 132:255 133:63 157:96 158:205 159:251 160:253 161:205 162
1 159:121 160:254 161:136 186:13 187:230 188:253 189:248 190:99 213:4 214:
1 100:166 101:222 102:55 128:197 129:254 130:218 131:5 155:29 156:249 157:254 158:254 159
0 155:53 156:255 157:253 158:253 159:253 160:124 183:180 184:253 185:251 186:251 187:251 188:251
0 128:73 129:253 130:227 131:73 132:21 156:73 157:251 158:251 159:251 160:174 161:16 183:166 184
1 155:178 156:255 157:185 182:6 183:188 184:253 185:216 186:14 210:11 211:202 212:253 213:253 21
0 154:46 155:105 156:254 157:254 158:254 159:254 160:254 161:254 162:41 180:37 181:118 182:222 1
0 152:56 153:105 154:220 155:254 156:254 157:254 179:166 180:233 181:253 182:253 183:253 184:236 1
0 152:76 153:224 133:224 158:51 159:253 160:253 161:233 185:4 186:170 187:253 188:253 189
0 155:21 156:176 157:253 158:253 159:253 160:253 161:253 162:253 163:253 164:253 165:253 166:253 167:253 168:253 169:253 170:253 171:253 172:253 173:253 174:253 175:253 176:253 177:253 178:253 179:253 180:253 181:253 182:253 183:253 184:253 185:253 186:253 187:253 188:253 189:253 190:253 191:253 192:253 193:253 194:253 195:253 196:253 197:253 198:253 199:253 200:253 201:253 202:253 203:253 204:253 205:253 206:253 207:253 208:253 209:253 210:253 211:253 212:253 213:253 214:253 215:253 216:253 217:253 218:253 219:253 220:253 221:253 222:253 223:253 224:253 225:253 226:253 227:253 228:253 229:253 230:253 231:253 232:253 233:253 234:253 235:253 236:253 237:253 238:253 239:253 240:253 241:253 242:253 243:253 244:253 245:253 246:253 247:253 248:253 249:253 250:253 251:253 252:253 253:253 254:253 255:253 256:253 257:253 258:253 259:253 260:253 261:253 262:253 263:253 264:253 265:253 266:253 267:253 268:253 269:253 270:253 271:253 272:253 273:253 274:253 275:253 276:253 277:253 278:253 279:253 280:253 281:253 282:253 283:253 284:253 285:253 286:253 287:253 288:253 289:253 290:253 291:253 292:253 293:253 294:253 295:253 296:253 297:253 298:253 299:253 300:253 301:253 302:253 303:253 304:253 305:253 306:253 307:253 308:253 309:253 310:253 311:253 312:253 313:253 314:253 315:253 316:253 317:253 318:253 319:253 320:253 321:253 322:253 323:253 324:253 325:253 326:253 327:253 328:253 329:253 330:253 331:253 332:253 333:253 334:253 335:253 336:253 337:253 338:253 339:253 340:253 341:253 342:253 343:253 344:253 345:253 346:253 347:253 348:253 349:253 350:253 351:253 352:253 353:253 354:253 355:253 356:253 357:253 358:253 359:253 360:253 361:253 362:253 363:253 364:253 365:253 366:253 367:253 368:253 369:253 370:253 371:253 372:253 373:253 374:253 375:253 376:253 377:253 378:253 379:253 380:253 381:253 382:253 383:253 384:253 385:253 386:253 387:253 388:253 389:253 390:253 391:253 392:253 393:253 394:253 395:253 396:253 397:253 398:253 399:253 400:253 401:253 402:253 403:253 404:253 405:253 406:253 407:253 408:253 409:253 410:253 411:253 412:253 413:253 414:253 415:253 416:253 417:253 418:253 419:253 420:253 421:253 422:253 423:253 424:253 425:253 426:253 427:253 428:253 429:253 430:253 431:253 432:253 433:253 434:253 435:253 436:253 437:253 438:253 439:253 440:253 441:253 442:253 443:253 444:253 445:253 446:253 447:253 448:253 449:253 450:253 451:253 452:253 453:253 454:253 455:253 456:253 457:253 458:253 459:253 460:253 461:253 462:253 463:253 464:253 465:253 466:253 467:253 468:253 469:253 470:253 471:253 472:253 473:253 474:253 475:253 476:253 477:253 478:253 479:253 480:253 481:253 482:253 483:253 484:253 485:253 486:253 487:253 488:253 489:253 490:253 491:253 492:253 493:253 494:253 495:253 496:253 497:253 498:253 499:253 500:253 501:253 502:253 503:253 504:253 505:253 506:253 507:253 508:253 509:253 510:253 511:253 512:253 513:253 514:253 515:253 516:253 517:253 518:253 519:253 520:253 521:253 522:253 523:253 524:253 525:253 526:253 527:253 528:253 529:253 530:253 531:253 532:253 533:253 534:253 535:253 536:253 537:253 538:253 539:253 540:253 541:253 542:253 543:253 544:253 545:253 546:253 547:253 548:253 549:253 550:253 551:253 552:253 553:253 554:253 555:253 556:253 557:253 558:253 559:253 560:253 561:253 562:253 563:253 564:253 565:253 566:253 567:253 568:253 569:253 570:253 571:253 572:253 573:253 574:253 575:253 576:253 577:253 578:253 579:253 580:253 581:253 582:253 583:253 584:253 585:253 586:253 587:253 588:253 589:253 590:253 591:253 592:253 593:253 594:253 595:253 596:253 597:253 598:253 599:253 600:253 601:253 602:253 603:253 604:253 605:253 606:253 607:253 608:253 609:253 610:253 611:253 612:253 613:253 614:253 615:253 616:253 617:253 618:253 619:253 620:253 621:253 622:253 623:253 624:253 625:253 626:253 627:253 628:253 629:253 630:253 631:253 632:253 633:253 634:253 635:253 636:253 637:253 638:253 639:253 640:253 641:253 642:253 643:253 644:253 645:253 646:253 647:253 648:253 649:253 650:253 651:253 652:253 653:253 654:253 655:253 656:253 657:253 658:253 659:253 660:253 661:253 662:253 663:253 664:253 665:253 666:253 667:253 668:253 669:253 670:253 671:253 672:253 673:253 674:253 675:253 676:253 677:253 678:253 679:253 680:253 681:253 682:253 683:253 684:253 685:253 686:253 687:253 688:253 689:253 690:253 691:253 692:253 693:253 694:253 695:253 696:253 697:253 698:253 699:253 700:253 701:253 702:253 703:253 704:253 705:253 706:253 707:253 708:253 709:253 710:253 711:253 712:253 713:253 714:253 715:253 716:253 717:253 718:253 719:253 720:253 721:253 722:253 723:253 724:253 725:253 726:253 727:253 728:253 729:253 730:253 731:253 732:253 733:253 734:253 735:253 736:253 737:253 738:253 739:253 740:253 741:253 742:253 743:253 744:253 745:253 746:253 747:253 748:253 749:253 750:253 751:253 752:253 753:253 754:253 755:253 756:253 757:253 758:253 759:253 760:253 761:253 762:253 763:253 764:253 765:253 766:253 767:253 768:253 769:253 770:253 771:253 772:253 773:253 774:253 775:253 776:253 777:253 778:253 779:253 780:253 781:253 782:253 783:253 784:253 785:253 786:253 787:253 788:253 789:253 790:253 791:253 792:253 793:253 794:253 795:253 796:253 797:253 798:253 799:253 800:253 801:253 802:253 803:253 804:253 805:253 806:253 807:253 808:253 809:253 810:253 811:253 812:253 813:253 814:253 815:253 816:253 817:253 818:253 819:253 820:253 821:253 822:253 823:253 824:253 825:253 826:253 827:253 828:253 829:253 830:253 831:253 832:253 833:253 834:253 835:253 836:253 837:253 838:253 839:253 840:253 841:253 842:253 843:253 844:253 845:253 846:253 847:253 848:253 849:253 850:253 851:253 852:253 853:253 854:253 855:253 856:253 857:253 858:253 859:253 860:253 861:253 862:253 863:253 864:253 865:253 866:253 867:253 868:253 869:253 870:253 871:253 872:253 873:253 874:253 875:253 876:253 877:253 878:253 879:253 880:253 881:253 882:253 883:253 884:253 885:253 886:253 887:253 888:253 889:253 890:253 891:253 892:253 893:253 894:253 895:253 896:253 897:253 898:253 899:253 900:253 901:253 902:253 903:253 904:253 905:253 906:253 907:253 908:253 909:253 910:253 911:253 912:253 913:253 914:253 915:253 916:253 917:253 918:253 919:253 920:253 921:253 922:253 923:253 924:253 925:253 926:253 927:253 928:253 929:253 930:253 931:253 932:253 933:253 934:253 935:253 936:253 937:253 938:253 939:253 940:253 941:253 942:253 943:253 944:253 945:253 946:253 947:253 948:253 949:253 950:253 951:253 952:253 953:253 954:253 955:253 956:253 957:253 958:253 959:253 960:253 961:253 962:253 963:253 964:253 965:253 966:253 967:253 968:253 969:253 970:253 971:253 972:253 973:253 974:253 975:253 976:253 977:253 978:253 979:253 980:253 981:253 982:253 983:253 984:253 985:253 986:253 987:253 988:253 989:253 990:253 991:253 992:253 993:253 994:253 995:253 996:253 997:253 998:253 999:253 1000:253
```

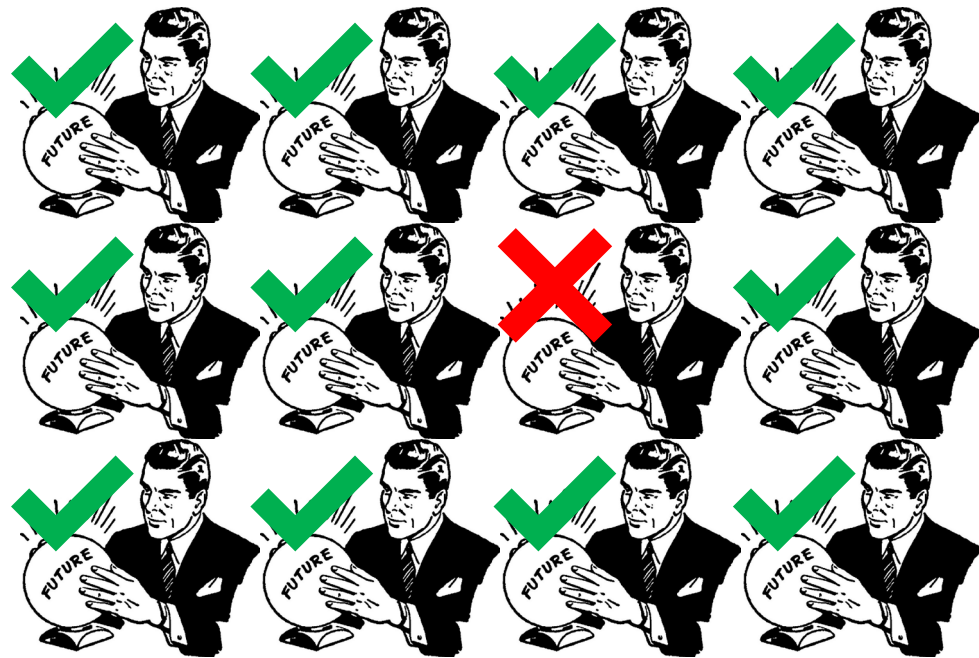
Datos

Predicción

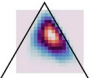
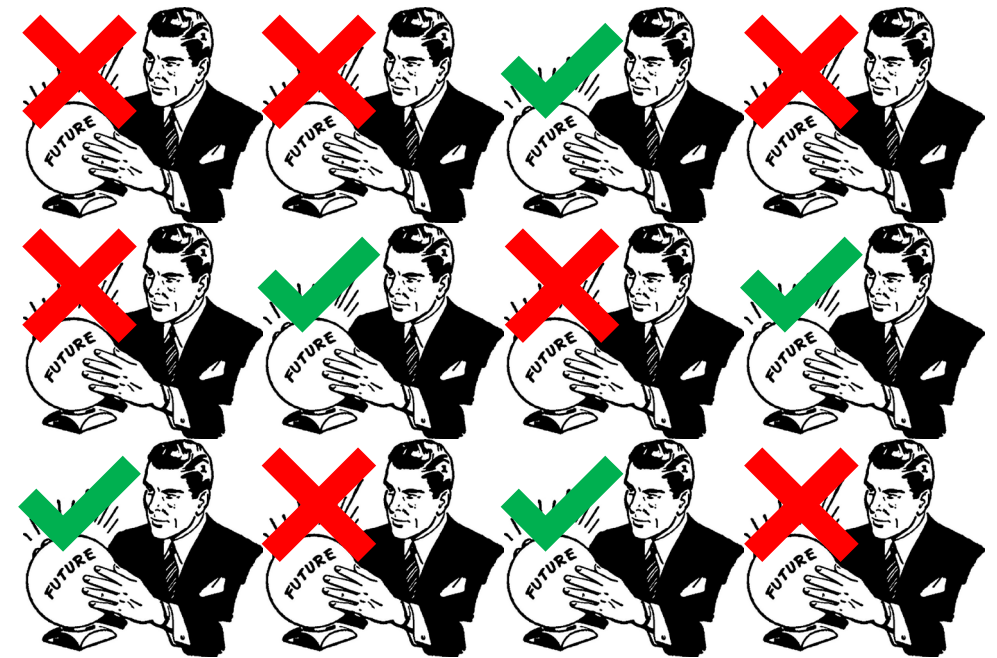
Veredicto



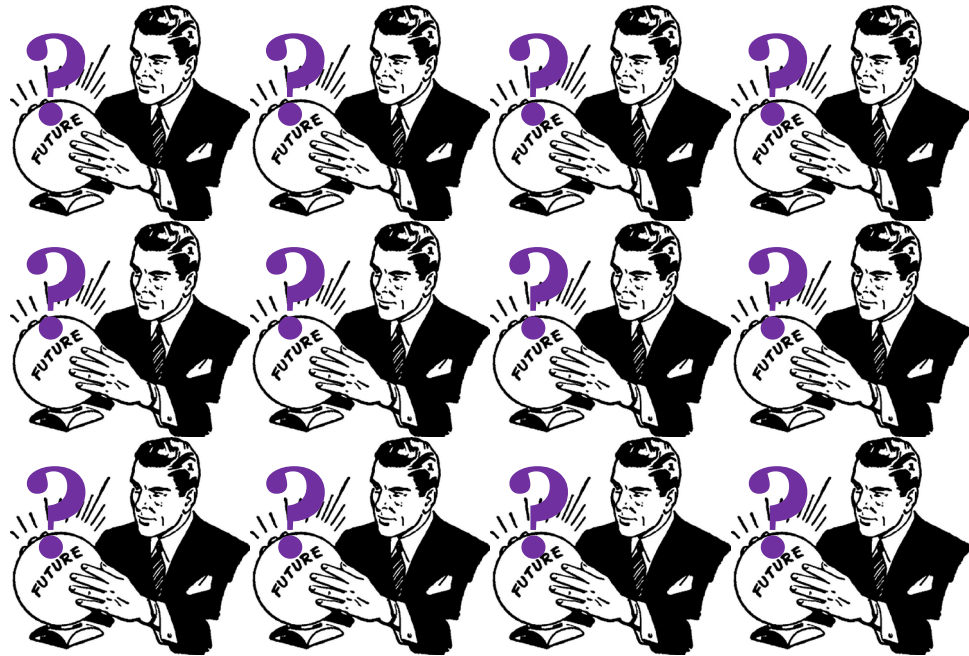
Mi clasificador 😎



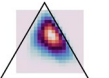
Tu clasificador 😞



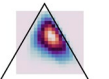
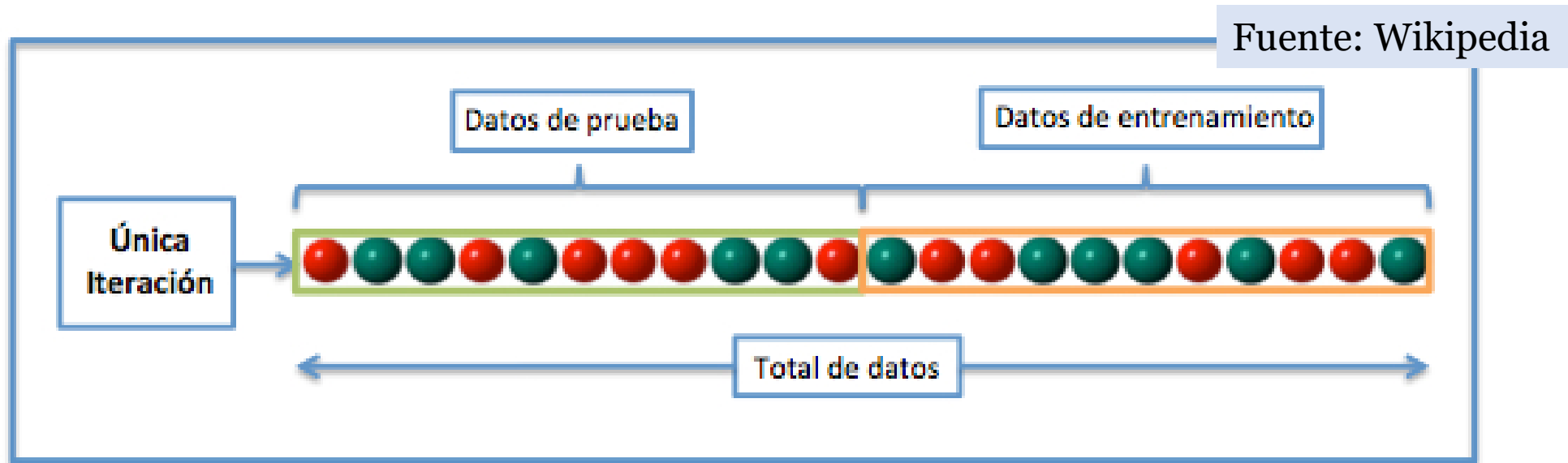
Mi clasificador



Tu clasificador

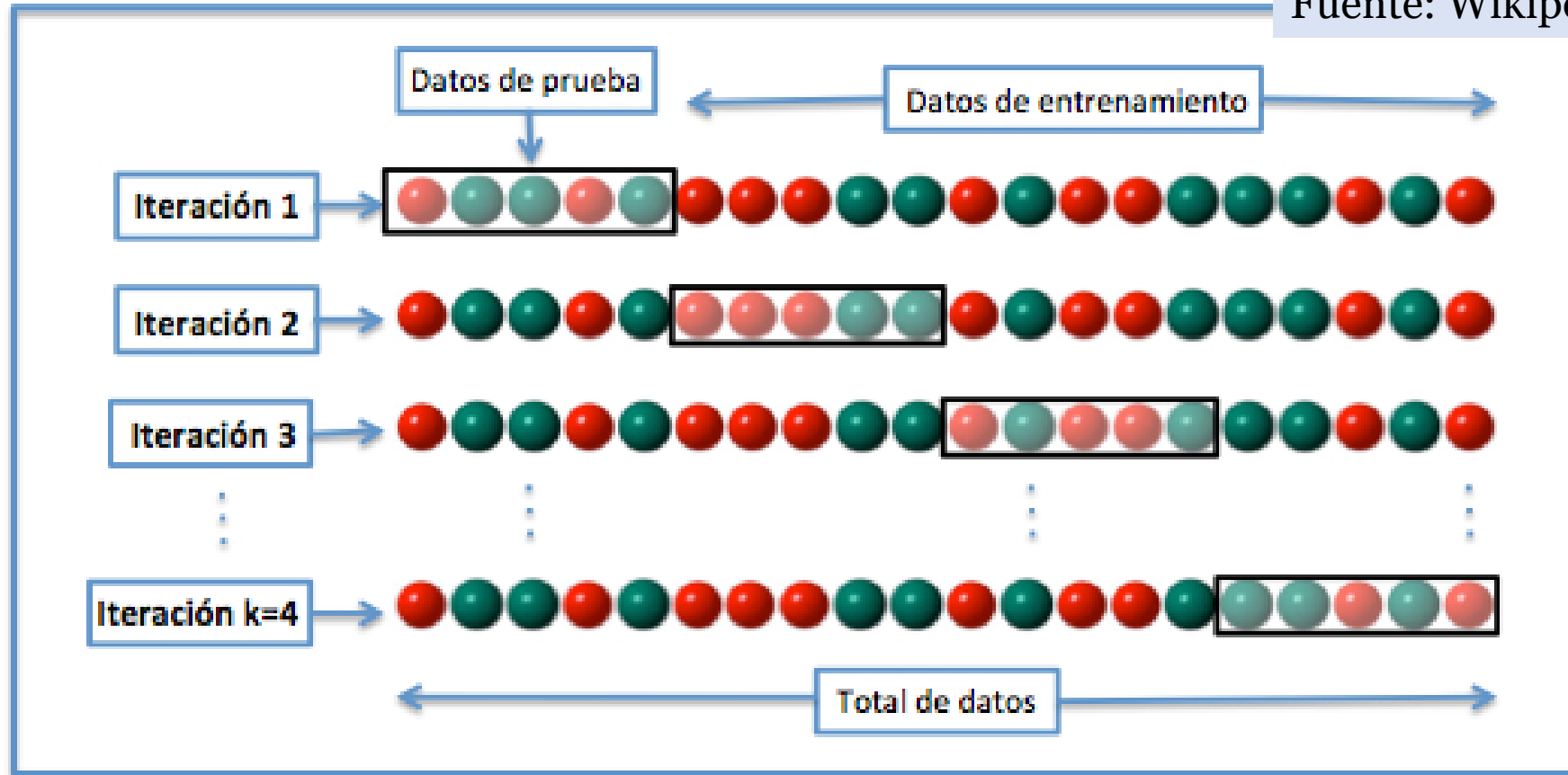


División de datos de entrenamiento

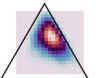


Validación cruzada

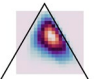
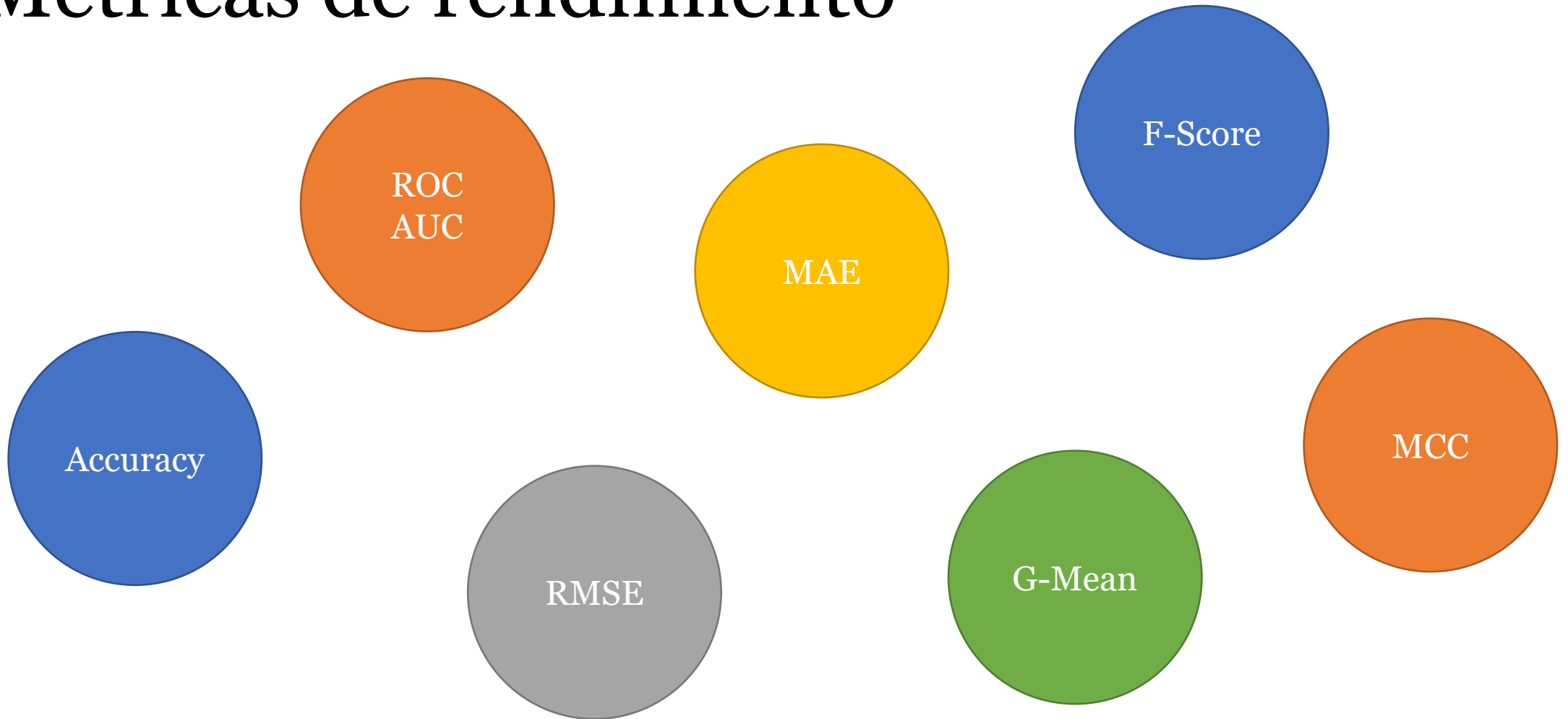
Fuente: Wikipedia



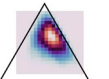
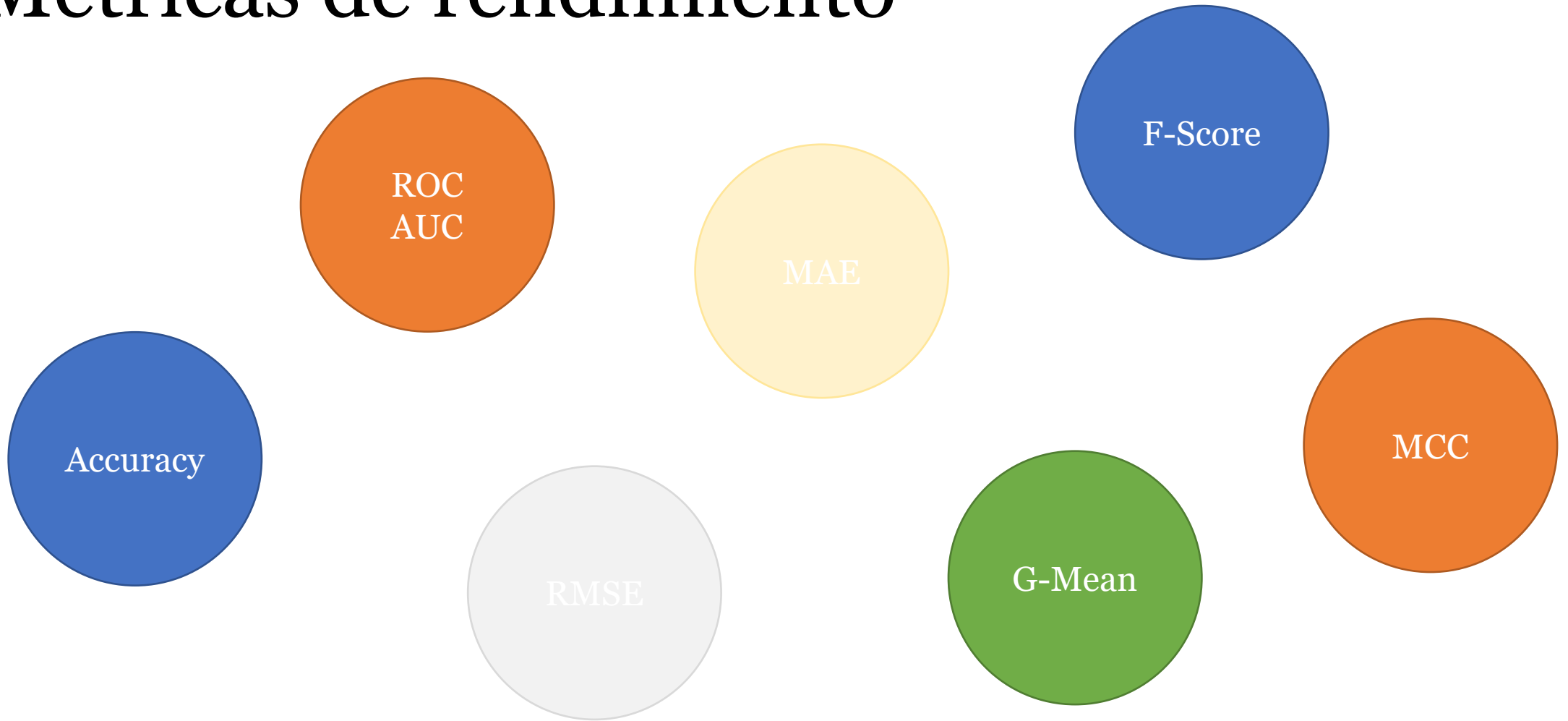
Métricas de rendimiento



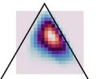
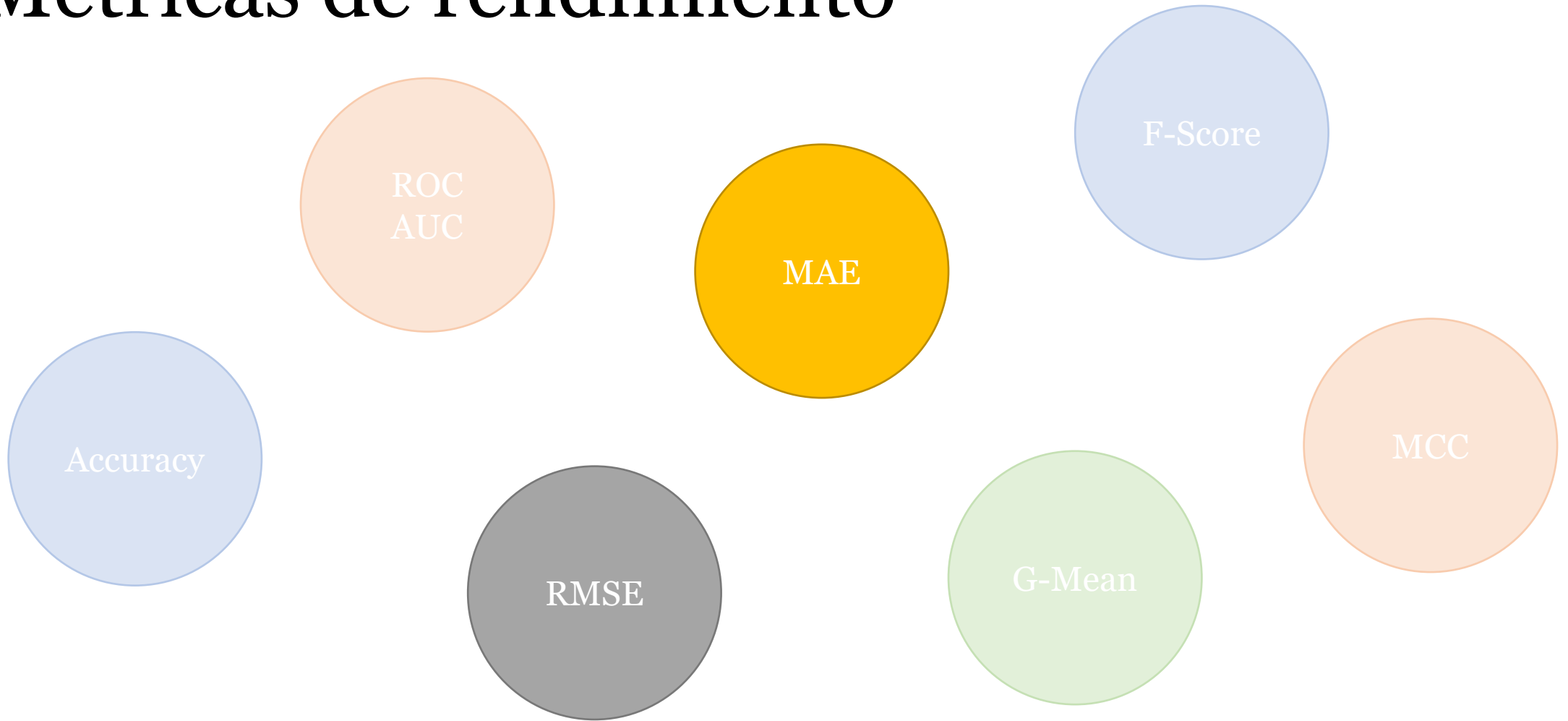
Métricas de rendimiento



Métricas de rendimiento



Métricas de rendimiento



Métricas de rendimiento

Accuracy

ROC
AUC

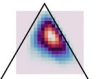
MAE

F-Score

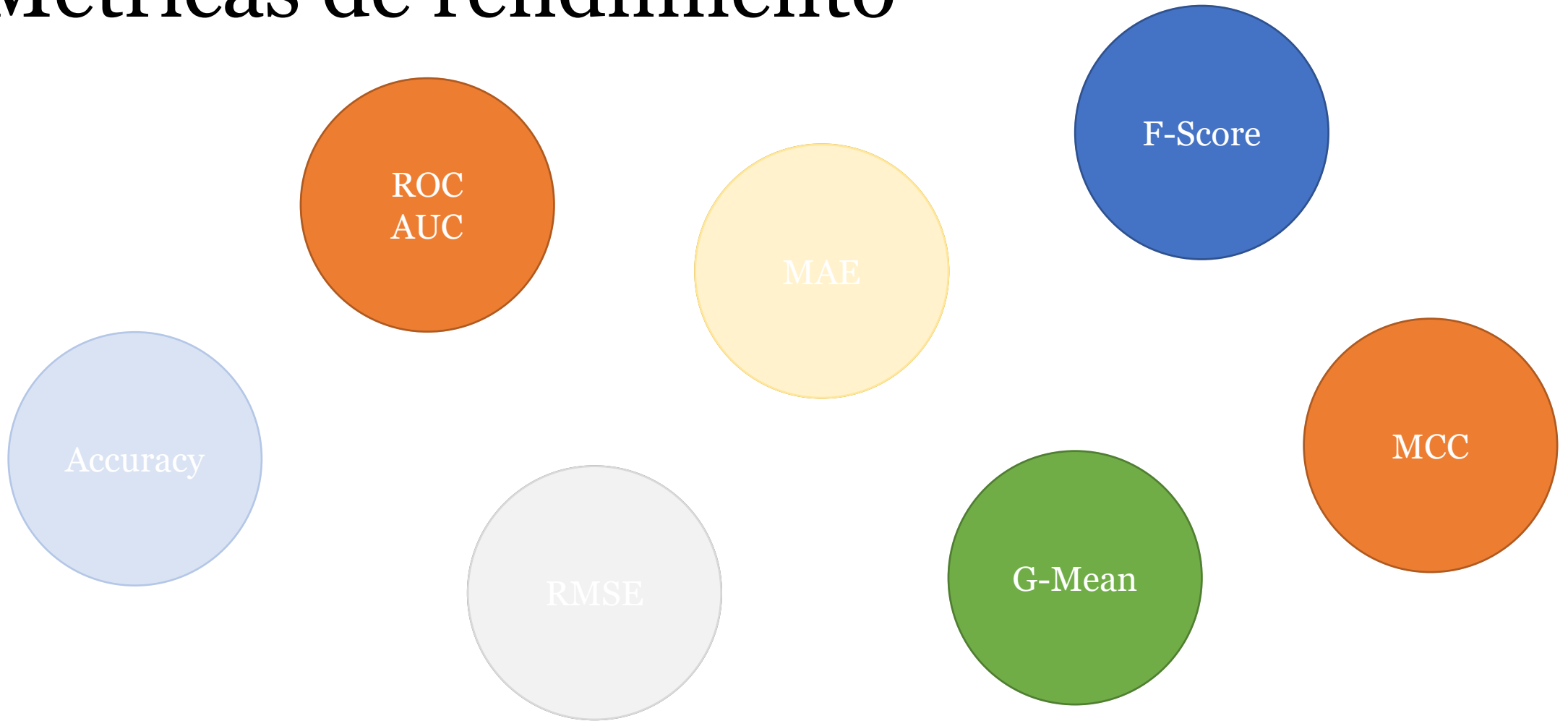
RMSE

G-Mean

MCC



Métricas de rendimiento



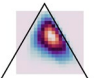
Comparación de clasificadores

Mi clasificador

FOLD	ACCURACY
1	0,68
2	0,72
3	0,88
4	0,61
5	0,75
media	0,73 ± 0,1

Tu clasificador

FOLD	ACCURACY
1	0,63
2	0,75
3	0,76
4	0,64
5	0,69
media	0,69 ± 0,06



Comparación de clasificadores

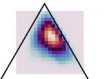
Mi clasificador 😎

FOLD	ACCURACY
1	0,68
2	0,72
3	0,88
4	0,61
5	0,75
media	0,73 ± 0,1

Tu clasificador 😞

FOLD	ACCURACY
1	0,63
2	0,75
3	0,76
4	0,64
5	0,69
media	0,69 ± 0,06

$$0,73 > 0,69$$



Comparación de clasificadores

Mi clasificador 😎 Tu clasificador 😞

FOLD	ACCURACY
1	0,68
2	0,72
3	0,88
4	0,61
5	0,75
media	0,73 ± 0,1

FOLD	ACCURACY
1	0,63
2	0,75
3	0,76
4	0,64
5	0,69
media	0,69 ± 0,06

- ¿Un conjunto de datos es suficiente?

0,73 > 0,69

Comparación de clasificadores

Mi clasificador 😎 Tu clasificador 😞

FOLD	ACCURACY
1	0,68
2	0,72
3	0,88
4	0,61
5	0,75
media	0,73 ± 0,1

FOLD	ACCURACY
1	0,63
2	0,75
3	0,76
4	0,64
5	0,69
media	0,69 ± 0,06

- ¿Un conjunto de datos es suficiente?
- ¿Diferencia significativa?

0,73 > 0,69

Comparación de clasificadores

Mi clasificador 😎 Tu clasificador 😞

FOLD	ACCURACY	FOLD	ACCURACY
1	0,68	1	0,63
2	0,72	2	0,75
3	0,88	3	0,76
4	0,61	4	0,64
5	0,75	5	0,69
media	0,73 ± 0,1	media	0,69 ± 0,06

0,73 > 0,69

- ¿Un conjunto de datos es suficiente?
- ¿Diferencia significativa?

Comparación estadística de clasificadores

2006

2008

2017

Journal of Machine Learning Research 7 (2006) 1–30

Submitted 8/04; Revised 4/05; Published 1/06

Statistical Comparisons of Classifiers over Multiple Data Sets

Janez Demšar

Faculty of Computer and Information Science
Tržaška 25
Ljubljana, Slovenia

JANEZ.DEMSAR@FRI.UNI-LJ.SI

Editor: Dale Schuurmans

Abstract

While methods for comparing two learning algorithms on a single data set have been scrutinized for quite some time already, the issue of statistical tests for comparisons of more algorithms on multiple data sets, which is even more essential to typical machine learning studies, has been all but ignored. This article reviews the current practice and then theoretically and empirically examines several suitable tests. Based on that, we recommend a set of simple, yet safe and robust non-parametric tests for statistical comparisons of classifiers: the Wilcoxon signed ranks test for comparison of two classifiers and the Friedman test with the corresponding post-hoc tests for comparison of more classifiers over multiple data sets. Results of the latter can also be neatly presented with the newly introduced CD (critical difference) diagrams.

Keywords: comparative studies, statistical methods, Wilcoxon signed ranks test, Friedman test, multiple comparisons tests

On the Appropriateness of Statistical Tests in Machine Learning

Janez Demšar

University of Ljubljana, Faculty of Computer Science, Tržaška 25, Ljubljana, Slovenia

JANEZ.DEMSAR@FRI.UNI-LJ.SI

Abstract

One of the greatest machine learning problems of today is an intractable number of new algorithms being presented at our conferences, workshops and journals. A similar rush of ideas and results also plagues many other scientific fields and some have already questioned the usefulness of statistical tests for telling the true relations from the false. Statistical tests have been criticized as conceptually wrong almost from their inception. They do not work well in situations when numerous groups conduct similar research. Not measuring what we are really interested in, they can promote the randomly successful ideas instead of the good but unlucky ones. We unfortunately see no established alternatives in other fields of science which could be transplanted to our field. We however speculate on a possible spontaneously appearing solution in a form of a worldwide peer review.

Machine learning is being suffocated by the ease with which we can generate new algorithms or, in most cases, slight variations of the existing ones by using flexible and extendible frameworks such as Weka (Witten & Frank, 2000) and many others. Our confes-

successful in beating the competition, although usually only marginally.

This perpetual enhancement of our methods – often without the new methods actually getting any wide attention and use after being published – should be suspicious and alarming by itself. Instead, we are getting used to read and hear about new and statistically significantly better methods... and do not pay any attention to them.

This pessimistic paper will first discuss why null-hypothesis significance testing is problematic in principle, next section will show why it is becoming inapplicable in most modern science including ours, followed and concluded by a section presenting several non-viable alternatives. The basic message of the paper is, however, that any evaluations and comparisons – statistical or non-statistical – of new methods should be taken with a grain of salt (as well as this paper itself).

1. Objections to Significance Tests

Null-hypothesis significance testing (NHST) is aimed at distinguishing between random and non-random differences or, in general, relations found in experimental results. Yet its appropriateness has been disputed from its beginnings. The heaviest fire comes

Journal of Machine Learning Research 18 (2017) 1–36

Submitted 6/16; Revised 5/17; Published 8/17

Time for a Change: a Tutorial for Comparing Multiple Classifiers Through Bayesian Analysis

Alessio Benavoli¹

Giorgio Corani¹

Janez Demšar²

Marco Zaffalon¹

¹Faculty of Computer and Information Science, University of Ljubljana, Vecna pot 113, SI-1000 Ljubljana, Slovenia
[†]Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA) Galleria 2, 6928 Manno, Switzerland

ALESSIO@IDSIA.CH

GIORGIO@IDSIA.CH

JANEZ.DEMSAR@FRI.UNI-LJ.SI

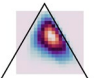
ZAFFALON@IDSIA.CH

Editor: David Barber

Abstract

The machine learning community adopted the use of null hypothesis significance testing (NHST) in order to ensure the statistical validity of results. Many scientific fields however realized the shortcomings of frequentist reasoning and in the most radical cases even banned its use in publications. We should do the same: just as we have embraced the Bayesian paradigm in the development of new machine learning methods, so we should also use it in the analysis of our own results. We argue for abandonment of NHST by exposing its fallacies and, more importantly, offer better—more sound and useful—alternatives for it.

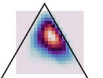
Keywords: comparing classifiers, null hypothesis significance testing, pitfalls of p-values, Bayesian hypothesis tests, Bayesian correlated t-test, Bayesian hierarchical correlated t-test, Bayesian signed-rank test



Comparación estadística de clasificadores

Antes de 2006:

- t-test
 - Test paramétrico
 - Muestras deben seguir una distribución normal ○
tamaño de la muestra: +30
 - Sensible a outliers (datos anómalos bajan fiabilidad del test)



Comparación estadística de clasificadores

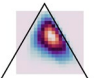
Antes de 2006:

- t-test
 - Test paramétrico
 - Muestras de una distribución normal
 - Se asume que los datos siguen una distribución normal (datos no normales bajan fiabilidad del test)

**NO SON
ADECUADOS**

A partir de 2006:

- Tests no paramétricos
 - Las muestras no deben ajustarse a distribuciones
 - Basados en rankings
 - Ej: Wilcoxon Signed-Rank

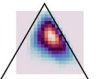


Comparación estadística de clasificadores

Contraste de hipótesis:

- H_0 (hipótesis nula): Rendimiento equivalente
- H_1 (hipótesis alternativa): El rendimiento no es equivalente

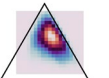
Rechazo de hipótesis \rightarrow mi clasificador es mejor que el tuyo



Wilcoxon Signed-Rank Test

Ranking de diferencias

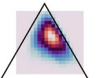
Dataset	Tu Clasificador	Mi Clasificador	Diferencia
adult (sample)	0.763	0.768	+0.005
breast cancer	0.599	0.591	-0.008
bc wisconsin	0.954	0.971	+0.017
cmc	0.628	0.661	+0.033
ionosphere	0.882	0.888	+0.006
iris	0.936	0.931	-0.005
liver disorders	0.661	0.668	+0.007
lung cancer	0.583	0.583	0.000
lymphography	0.775	0.838	+0.063
mushroom	1.000	1.000	0.000
primary tumor	0.940	0.962	+0.022
rheum	0.619	0.666	+0.047
voting	0.972	0.981	+0.009
wine	0.957	0.978	+0.021



Wilcoxon Signed-Rank Test

Ranking de diferencias

Dataset	Tu Clasificador	Mi Clasificador	Diferencia	Rank
adult (sample)	0.763	0.768	+0.005	3.5
breast cancer	0.599	0.591	-0.008	7
bc wisconsin	0.954	0.971	+0.017	9
cmc	0.628	0.661	+0.033	12
ionosphere	0.882	0.888	+0.006	5
iris	0.936	0.931	-0.005	3.5
liver disorders	0.661	0.668	+0.007	6
lung cancer	0.583	0.583	0.000	1.5
lymphography	0.775	0.838	+0.063	14
mushroom	1.000	1.000	0.000	1.5
primary tumor	0.940	0.962	+0.022	11
rheum	0.619	0.666	+0.047	13
voting	0.972	0.981	+0.009	8
wine	0.957	0.978	+0.021	10



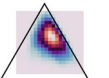
Wilcoxon Signed-Rank Test

Ranking de diferencias

Dataset	Tu Clasificador	Mi Clasificador	Diferencia	Rank
adult (sample)	0.763	0.768	+0.005	3.5
breast cancer	0.599	0.591	-0.008	7
bc wisconsin	0.954	0.971	+0.017	9
cmc	0.628	0.661	+0.033	12
ionosphere	0.882	0.888	+0.006	5
iris	0.936	0.931	-0.005	3.5
liver disorders	0.661	0.668	+0.007	6
lung cancer	0.583	0.583	0.000	1.5
lymphography	0.775	0.838	+0.063	14
mushroom	1.000	1.000	0.000	1.5
primary tumor	0.940	0.962	+0.022	11
rheum	0.619	0.666	+0.047	13
voting	0.972	0.981	+0.009	8
wine	0.957	0.978	+0.021	10

$$R_+ = 93$$

$$R_- = 12$$



Wilcoxon Signed-Rank Test

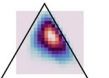
$$R_+ = 93, R_- = 12$$

$$T = \min(R_+, R_-) = 12$$

$$N = 14$$

Tabla de valores críticos

n	alpha values						
	0.001	0.005	0.01	0.025	0.05	0.10	0.20
5	--	--	--	--	--	0	2
6	--	--	--	--	0	2	3
7	--	--	--	0	2	3	5
8	--	--	0	2	3	5	8
9	--	0	1	3	5	8	10
10	--	1	3	5	8	10	14
11	0	3	5	8	10	13	17
12	1	5	7	10	13	17	21
13	2	7	9	13	17	21	26
14	4	9	12	17	21	25	31
15	6	12	15	20	25	30	36
16	8	15	19	25	29	35	42
17	11	19	23	29	34	41	48
18	14	23	27	34	40	47	55
19	18	27	32	39	46	53	62
20	21	32	37	45	52	60	69
21	25	37	42	51	58	67	77
22	30	42	48	57	65	75	86
23	35	48	54	64	73	83	94
24	40	54	61	72	81	91	104
25	45	60	68	79	89	100	113
26	51	67	75	87	98	110	124
27	57	74	83	96	107	119	134



Wilcoxon Signed-Rank Test

$$R_+ = 93, R_- = 12$$

$$T = \min(R_+, R_-) = 12$$

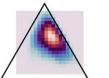
$$N = 14$$

$12 < 21$,
se rechaza H_0 ,

Mi clasificador es mejor que el tuyo

Tabla de valores críticos

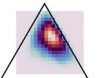
n	alpha values						
	0.001	0.005	0.01	0.025	0.05	0.10	0.20
5	--	--	--	--	--	0	2
6	--	--	--	--	0	2	3
7	--	--	--	0	2	3	5
8	--	--	0	2	3	5	8
9	--	0	1	3	5	8	10
10	--	1	3	5	8	10	14
11	0	3	5	8	10	13	17
12	1	5	7	10	13	17	21
13	2	7	9	13	17	21	26
14	4	9	12	17	21	25	31
15	6	12	15	20	25	30	36
16	8	15	19	25	29	35	42
17	11	19	23	29	34	41	48
18	14	23	27	34	40	47	55
19	18	27	32	39	46	53	62
20	21	32	37	45	52	60	69
21	25	37	42	51	58	67	77
22	30	42	48	57	65	75	86
23	35	48	54	64	73	83	94
24	40	54	61	72	81	91	104
25	45	60	68	79	89	100	113
26	51	67	75	87	98	110	124
27	57	74	83	96	107	119	134



Problemas del contraste de hipótesis

$$\alpha = 0.05$$

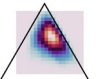
Mi clasificador es mejor que el tuyo con un 95% de probabilidad



Problemas del contraste de hipótesis

$$\alpha = 0.05$$

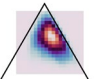
~~Mi clasificador es mejor que el tuyo con un 95% de probabilidad~~



Problemas del contraste de hipótesis

$$\alpha = 0.05$$

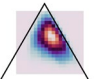
Hay un 5% de probabilidad de que dos clasificadores iguales presenten las mismas (o mayores) diferencias vistas en los experimentos.



Problemas del contraste de hipótesis

Pero yo quiero saber:

¿Con qué probabilidad mi clasificador es mejor que el tuyo?



Teorema de Bayes



Teorema de Bayes



Tu clasificador y
mi clasificador
son iguales

Resultados
experimentales

Mi clasificador es
mejor que el tuyo
(con un 80% de probabilidad)

Teorema de Bayes

$$P(A|E) = \frac{P(E|A)P(A)}{P(E)}$$

Probabilidad a posteriori

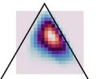
Probabilidad a priori

Evidencias

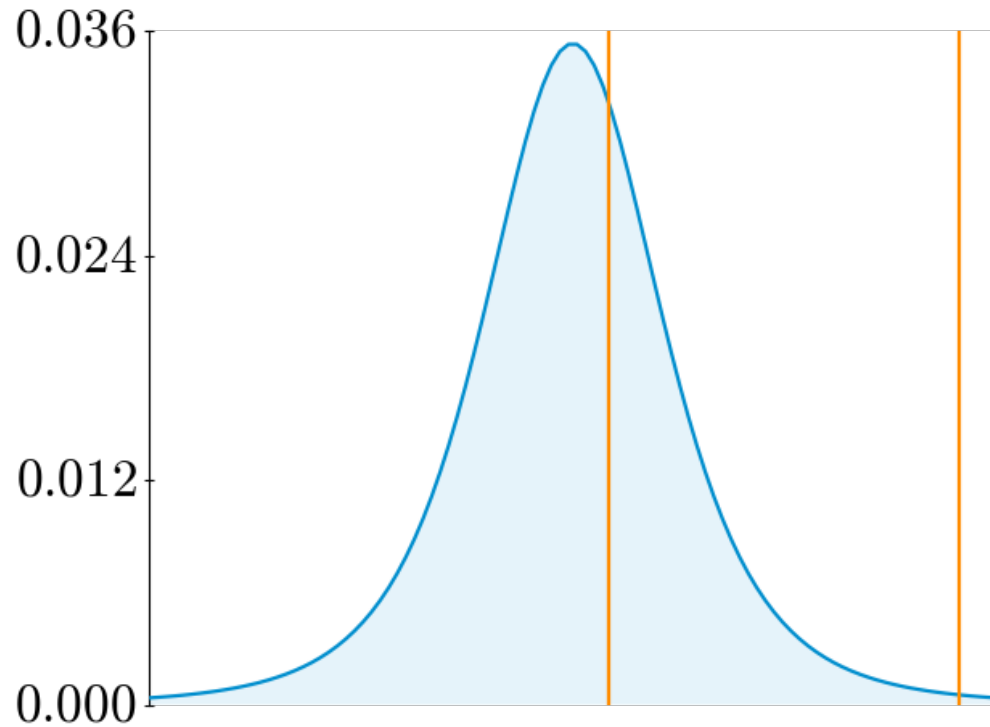


Comparación bayesiana de clasificadores

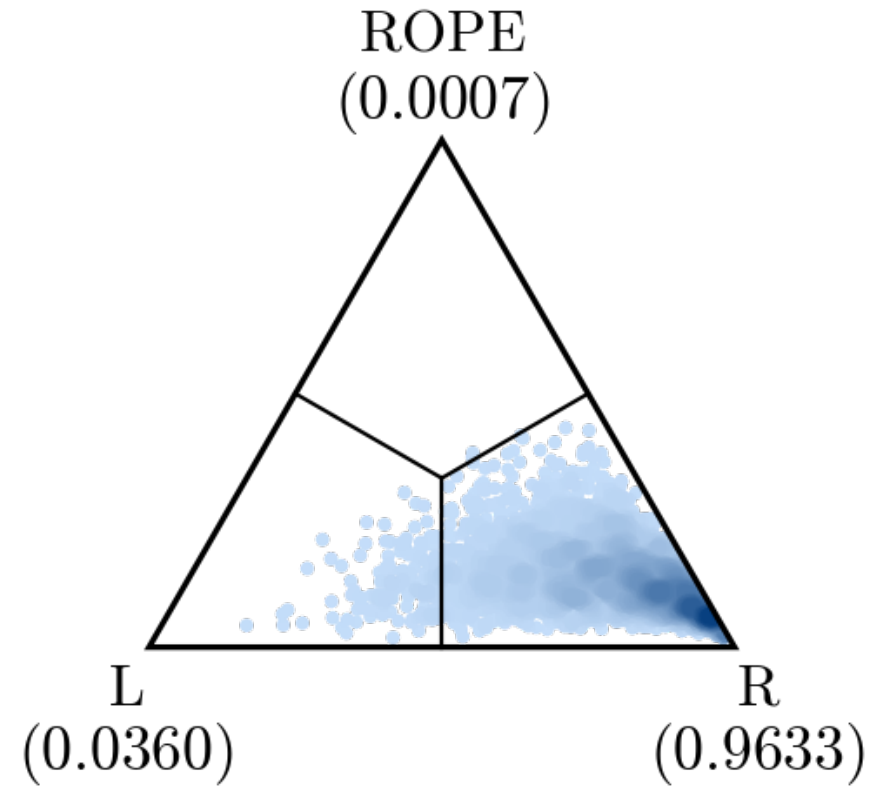
- Probabilidad de que mi clasificador sea mejor que el tuyo (L)
- Probabilidad de que tu clasificador sea mejor que el mío (R)
- Probabilidad de que ambos clasificadores rindan igual (ROPE)
 - Region Of Practical Equivalence



Comparación bayesiana de clasificadores



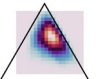
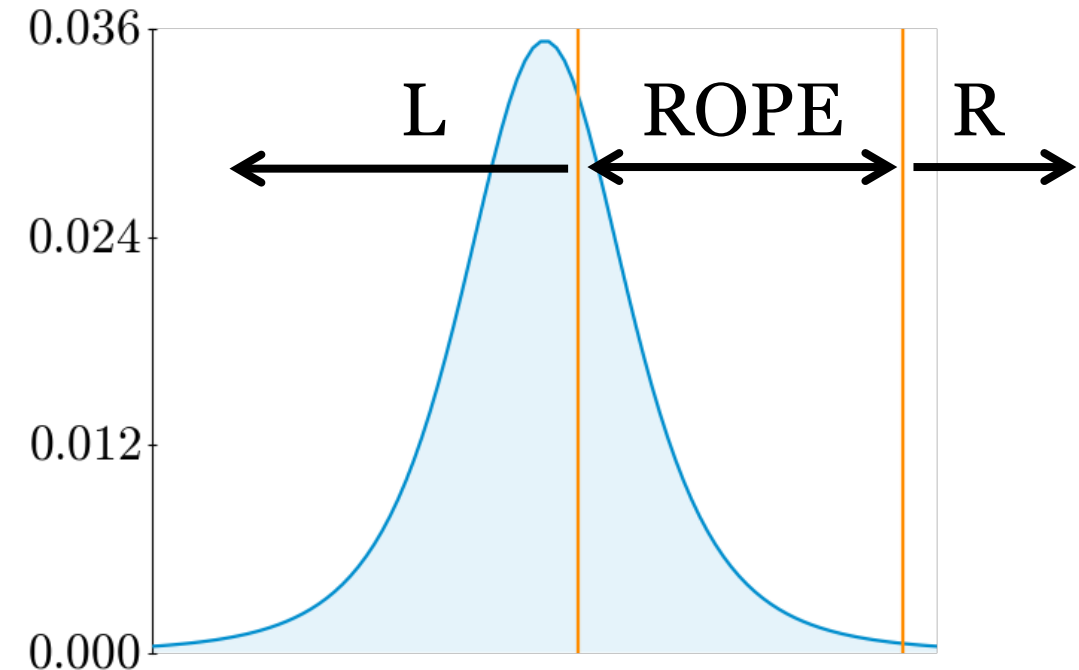
Un conjunto de datos



Varios conjuntos de datos

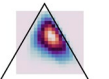
Un conjunto de datos

- Correlated t-test
- Necesario utilizar validación cruzada
- Distribución se calcula analíticamente



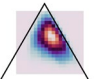
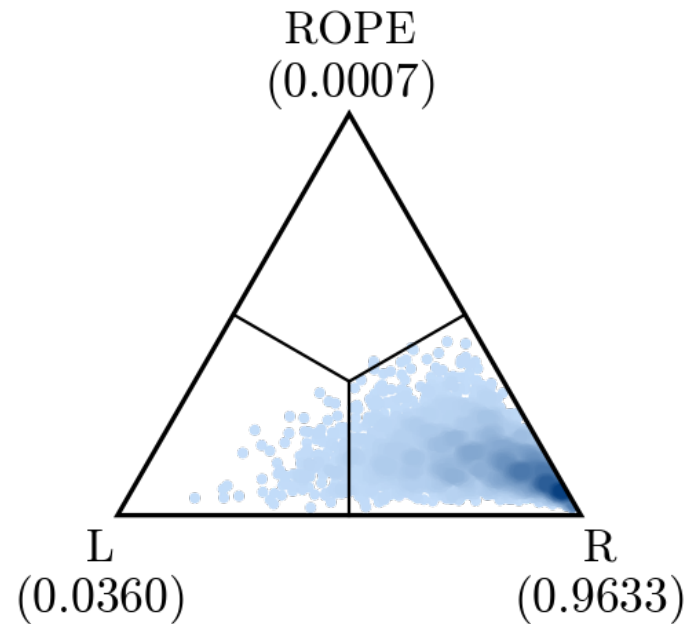
Varios conjuntos de datos

- Sign Test
 - Signed-Rank Test
- } Una medida por conjunto de datos
Versión bayesiana del test de Wilcoxon
- Hierarchical Test → Todas las medidas de todos los folds



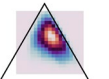
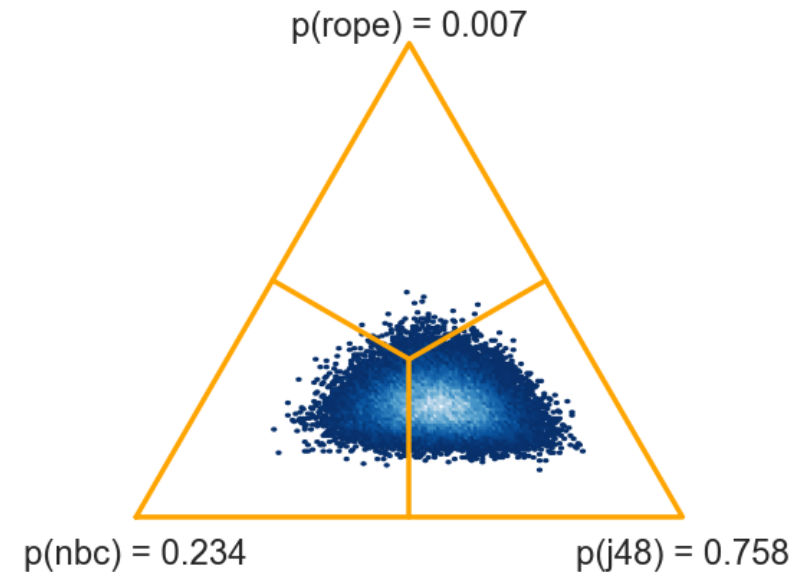
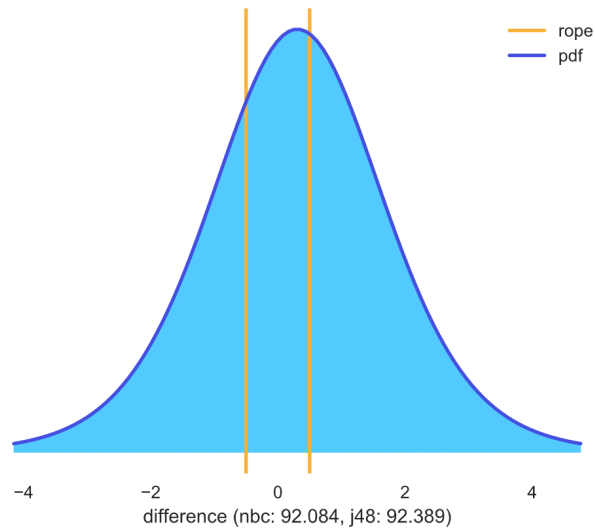
Varios conjuntos de datos

- Coste del cálculo analítico de la distribución muy elevado
- Se aproxima la distribución utilizando Monte Carlo



¿Cómo hago estas comparaciones?

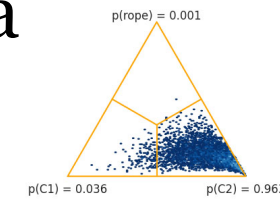
- Baycomp: Librería en Python
 - <https://baycomp.readthedocs.io>



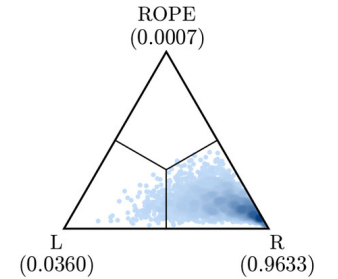
¿Cómo hago estas comparaciones?

- Baycomp_plotting: Visualización alternativa
 - https://github.com/mjuez/baycomp_plotting

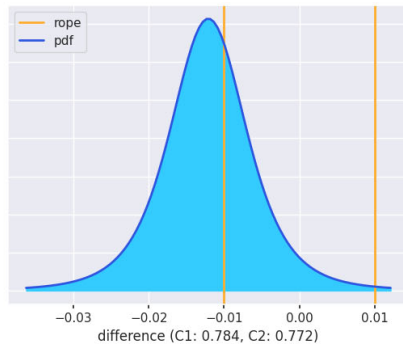
baycomp default plotting:



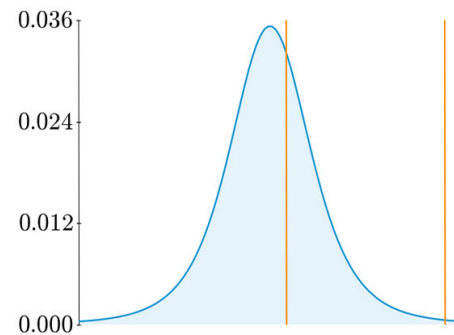
baycomp_plotting:



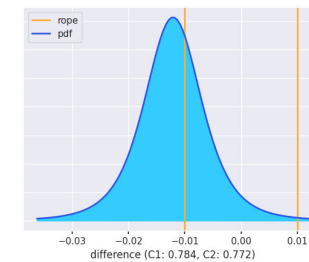
baycomp default plotting:



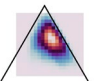
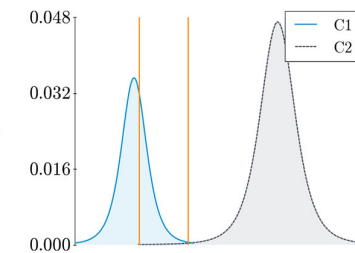
baycomp_plotting:



baycomp default plotting:

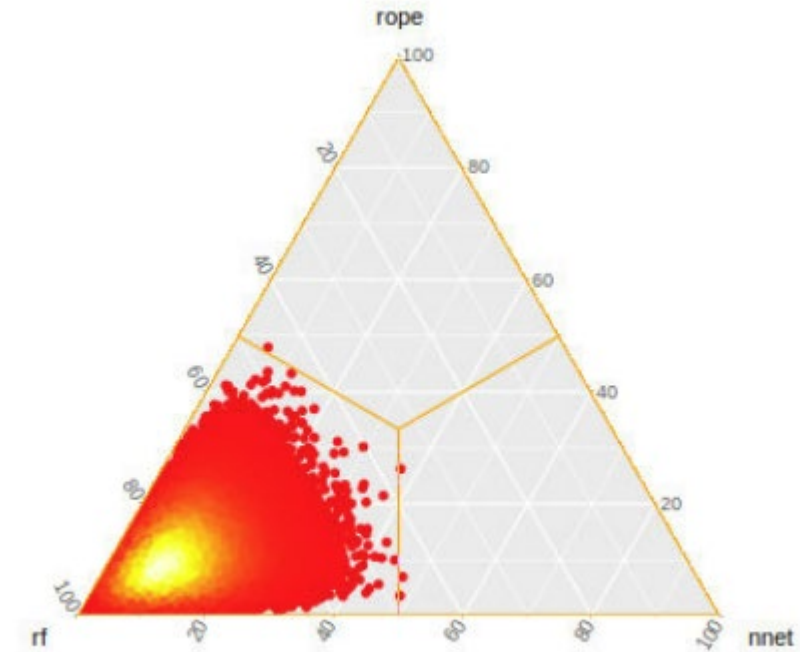
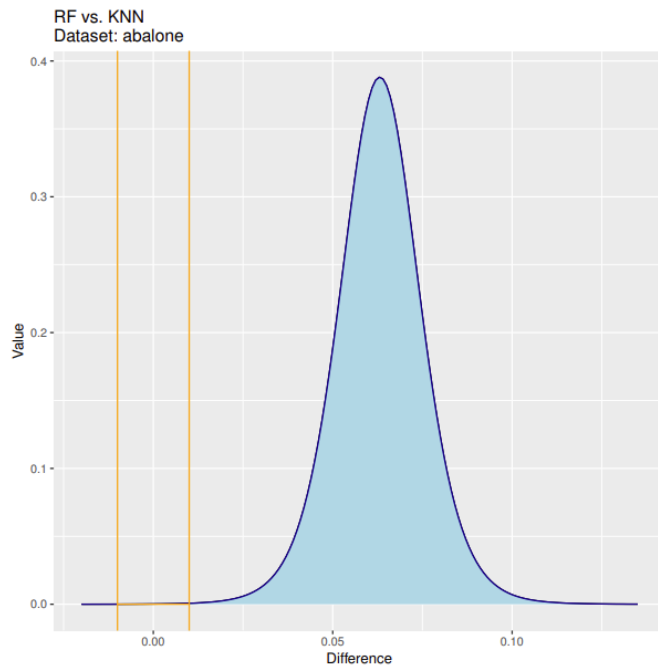


baycomp_plotting:



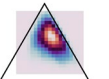
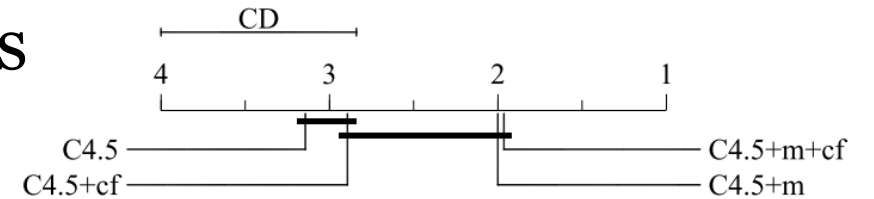
¿Cómo hago estas comparaciones?

- rNPBST: Librería en R
 - <https://jacintocc.github.io/rNPBST>



¿Hacia dónde vamos?

- Comparaciones de más de 2 clasificadores
 - Contraste de hipótesis: ANOVA, Friedmann
 - No existe alternativa bayesiana (aún)
- Resultados condicionados por los conjuntos de datos
 - Teorema: No Free Lunch
 - Mi clasificador es mejor que el tuyo... frente a estos datos concretos
- Alternativas al uso de métricas de rendimiento
 - Métodos gráficos



Referencias

- [Demšar, J. \(2006\). **Statistical comparisons of classifiers over multiple data sets.** *The Journal of Machine Learning Research*, 7, 1-30.](#)
- [Demšar, J. \(2008\). **On the appropriateness of statistical tests in machine learning.** In *Workshop on Evaluation Methods for Machine Learning in conjunction with ICML* \(p. 65\).](#)
- [Benavoli, A., Corani, G., Demšar, J., & Zaffalon, M. \(2017\). **Time for a change: a tutorial for comparing multiple classifiers through Bayesian analysis.** *The Journal of Machine Learning Research*, 18\(1\), 2653-2688.](#)
- [Carrasco, J., Garcia, S., & Herrera, F. **shinytests: Una herramienta gráfica para la comparacion estadística en minería de datos.**](#)
- [Stapor, K., Ksieniewicz, P., García, S., & Woźniak, M. \(2021\). **How to design the fair experimental classifier evaluation.** *Applied Soft Computing*, 104, 107219.](#)

