

1 Introduction

Processing the sparse data file format with Tanagra¹.

The data to be processed with machine learning algorithms are increasing in size. Especially when we need to process [unstructured data](#). The data preparation (e. g. the use of a bag of words representation in text mining) leads to the creation of large data tables where, often, the number of columns (descriptors) is higher than the number of rows (observations). With the singularity that the table contains many zero values. In this context, storing all these zero values into the data file is not opportune. A data compression strategy without loss of information must be implemented, which must remain simple so that the file is readable with a text editor.

In this tutorial, we describe the use of the sparse data file format handled by **Tanagra** (from the version 1.4.4). It is based on the file format processed by famous libraries for machine learning (svmlight, libsvm, libsvm)². We show its use in a text categorization process applied to the Reuters database, well known in data mining³. We will observe that the use of this kind of sparse format enables to reduce dramatically the data file size.

2 The “sparse” data file format

2.1 From attribute-value table format to the sparse format

In text mining, the starting point is always a document collection. But the machine learning algorithms cannot process the raw documents. We must transform the document collection in the attribute value table which is the usual data representation. The bag-of words model is definitely the most popular approach.

Let us consider an example to detail the approach. We have a collection of 3 documents (in French):

- A. “Le soleil brille dans le ciel”
- B. “Le ciel est bleu”
- C. “Voilà le produit pour faire briller”

¹ The French version of this tutorial is written in May 2012. Unfortunately, it seems that some websites links are broken. I did not find the updated links.

² <http://svmlight.joachims.org/> ; <http://www.csie.ntu.edu.tw/~cjlin/libsvm/> ; <http://c2inet.sce.ntu.edu.sg/ivor/cvm.html>

³ <http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html>

According to the bag-of-words model, we convert each term as a descriptor. The attribute-value table (3 rows/documents, 12 columns/descriptors/terms) corresponding to the document collection above is:

Document	1 le	2 soleil	3 brille	4 dans	5 ciel	6 est	7 bleu	8 voilà	9 produit	10 pour	11 faire	12 briller
A	2	1	1	1	1	0	0	0	0	0	0	0
B	1	0	0	0	1	1	1	0	0	0	0	0
C	1	0	0	0	0	0	0	1	1	1	1	1

Two characteristics stand out. (1) The data table can quickly be very large. Indeed, the columns correspond to the list of the words likely to appear in all the documents. Their number will be even higher as the size and the number of the documents increase. (2) There are many zero value into the table. Indeed, few words, among all the possible terms, appear in each document. It is not interesting to store this information explicitly in the data file. A compression strategy would be helpful.

The "sparse" format consists of identifying only the terms that actually appear in each document. For our example, considering that each word is associated with a number, we will store the data in the following form:

Document	Description
A	1:2 2:1 3:1 4:1 5:1
B	1:1 5:1 6:1 7:1
C	1:1 8:1 9:1 10:1 11:1 12:1

Each non-zero value is prefixed by its column number. On the other hand, columns that do not appear in the row implicitly correspond to the value 0. Thus, we hope to achieve a significant reduction in the space needed for storage. It will be all the more spectacular as the proportion of zero values is high in our data table.

2.2 The "sparse" for regression and classification

In the predictive analysis process, each observation is labelled either by a number indicating its membership if we are in a classification context, or by the value of the response variable if we are in a regression context. As for the SvmLight and Libsvm libraries, the label is placed first in the row.

For the document collection above, let us consider that the documents A and B belong to the first class "1", and C belongs to "-1". Our data file has the following appearance:

```
1 1:2 2:1 3:1 4:1 5:1
1 1:1 5:1 6:1 7:1
-1 1:1 8:1 9:1 10:1 11:1 12:1
```

Note: Whether in classification or regression, the label always corresponds to a numerical value in the file. A recoding will be necessary into Tanagra to consider that it is an indicator of the class membership.

2.3 The "reuters (money fx)" database

We process the Reuters database. It is a classic of text categorization problem. We have a collection of news. Each of them is indexed by one or more categories (TOPICS). The following document for example is associated to the topics "money-fx" and "interest".

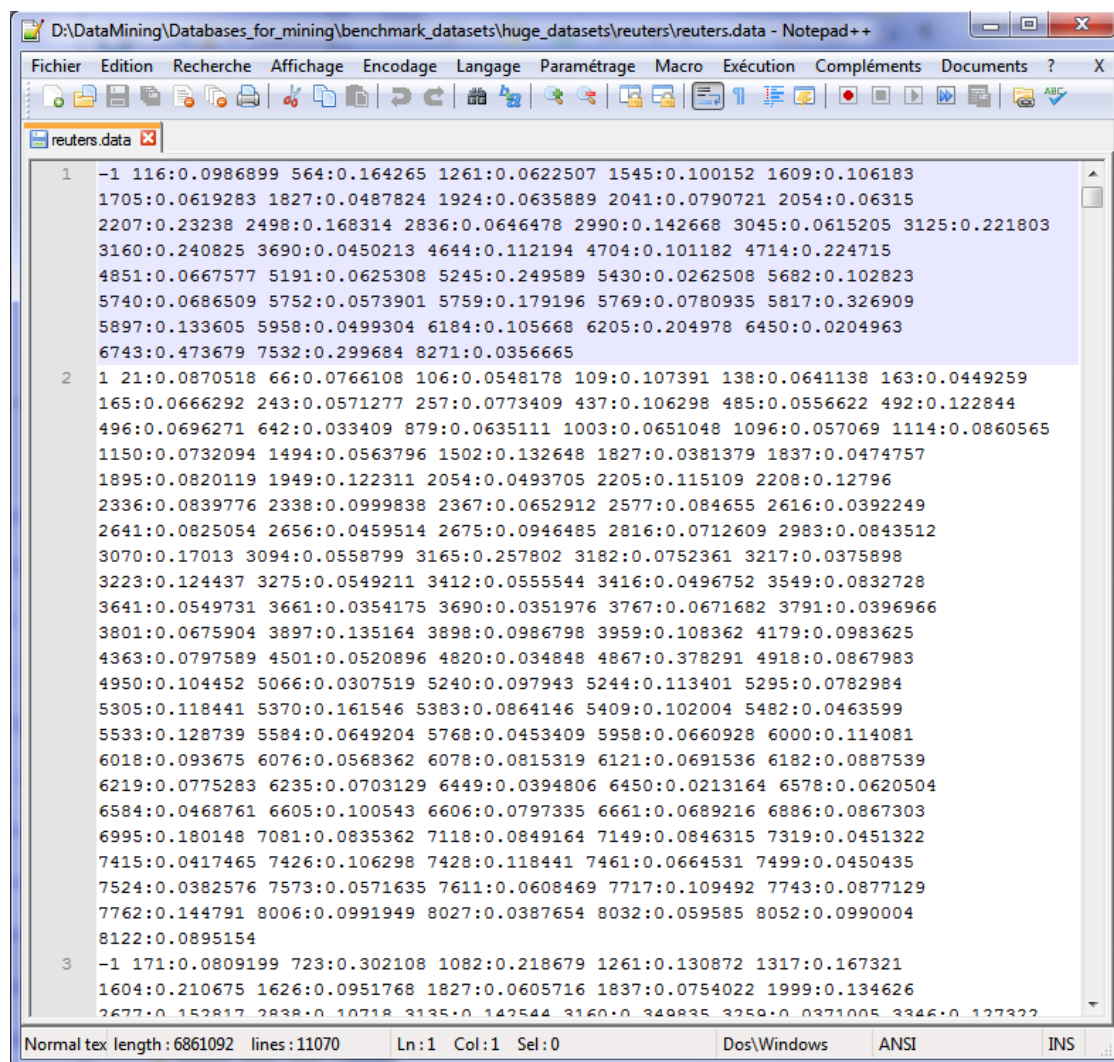
```
<REUTERS TOPICS="YES" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET" OLDID="5764" NEWID="221">
<DATE>26-FEB-1987 21:05:51.60</DATE>
<TOPICS><D>money-fx</D><D>interest</D></TOPICS>
<PLACES><D>japan</D></PLACES>
<PEOPLE></PEOPLE>
<ORGS></ORGS>
<EXCHANGES></EXCHANGES>
<COMPANIES></COMPANIES>
<UNKNOWN>
&#5;&#5;&#5;RM
&#22;&#22;&#1;f0438&#31;reute
u f BC-AVERAGE-YEN-CD-RATES 02-26 0096</UNKNOWN>
<TEXT>&#2;
<TITLE>AVERAGE YEN CD RATES FALL IN LATEST WEEK</TITLE>
<DATELINE> TOKYO, Feb 27 - </DATELINE><BODY>Average interest rates on yen certificates
of deposit, CD, fell to 4.27 pct in the week ended February 25
from 4.32 pct the previous week, the Bank of Japan said.
  New rates (previous in brackets), were -
  Average CD rates all banks 4.27 pct (4.32)
  Money Market Certificate, MMC, ceiling rates for the week
starting from March 2 3.52 pct (3.57)
  Average CD rates of city, trust and long-term banks
  Less than 60 days 4.33 pct (4.32)
  60-90 days 4.13 pct (4.37)
  Average CD rates of city, trust and long-term banks
  90-120 days 4.35 pct (4.30)
  120-150 days 4.38 pct (4.29)
  150-180 days unquoted (unquoted)
  180-270 days 3.67 pct (unquoted)
  Over 270 days 4.01 pct (unquoted)
  Average yen bankers' acceptance rates of city, trust and
long-term banks
  30 to less than 60 days unquoted (4.13)
  60-90 days unquoted (unquoted)
  90-120 days unquoted (unquoted)
  REUTER
&#3;</BODY></TEXT>
</REUTERS>
```

The objective of modeling is to learn from labelled data a classification function that automatically associates a new unseen instance to a class. In practice, we prefer to consider the treatment

from a binary perspective in order to simplify the process i.e. we want to know if a text belongs to a specific category. Into the learning set, "positive" documents are those related to the target category; "negative" documents are the others.

About the database handled in this tutorial, the target category is "**money-fx**". The transformation of the document collection in an attribute-value table was already done. We have 8315 descriptors. Initially, the dataset is composed of 7770 instances for the learning set, 3299 for the test set. They are merged in a unique data file [reuters.data](#).

Here are the first rows of the database.



```
1 -1 116:0.0986899 564:0.164265 1261:0.0622507 1545:0.100152 1609:0.106183
1705:0.0619283 1827:0.0487824 1924:0.0635889 2041:0.0790721 2054:0.06315
2207:0.23238 2498:0.168314 2836:0.0646478 2990:0.142668 3045:0.0615205 3125:0.221803
3160:0.240825 3690:0.0450213 4644:0.112194 4704:0.101182 4714:0.224715
4851:0.0667577 5191:0.0625308 5245:0.249589 5430:0.0262508 5682:0.102823
5740:0.0686509 5752:0.0573901 5759:0.179196 5769:0.0780935 5817:0.326909
5897:0.133605 5958:0.0499304 6184:0.105668 6205:0.204978 6450:0.0204963
6743:0.473679 7532:0.299684 8271:0.0356665

2 1 21:0.0870518 66:0.0766108 106:0.0548178 109:0.107391 138:0.0641138 163:0.0449259
165:0.0666292 243:0.0571277 257:0.0773409 437:0.106298 485:0.0556622 492:0.122844
496:0.0696271 642:0.033409 879:0.0635111 1003:0.0651048 1096:0.057069 1114:0.0860565
1150:0.0732094 1494:0.0563796 1502:0.132648 1827:0.0381379 1837:0.0474757
1895:0.0820119 1949:0.122311 2054:0.0493705 2205:0.115109 2208:0.12796
2336:0.0839776 2338:0.0999838 2367:0.0652912 2577:0.084655 2616:0.0392249
2641:0.0825054 2656:0.0459514 2675:0.0946485 2816:0.0712609 2983:0.0843512
3070:0.17013 3094:0.0558799 3165:0.257802 3182:0.0752361 3217:0.0375898
3223:0.124437 3275:0.0549211 3412:0.0555544 3416:0.0496752 3549:0.0832728
3641:0.0549731 3661:0.0354175 3690:0.0351976 3767:0.0671682 3791:0.0396966
3801:0.0675904 3897:0.135164 3898:0.0986798 3959:0.108362 4179:0.0983625
4363:0.0797589 4501:0.0520896 4820:0.034848 4867:0.378291 4918:0.0867983
4950:0.104452 5066:0.0307519 5240:0.097943 5244:0.113401 5295:0.0782984
5305:0.118441 5370:0.161546 5383:0.0864146 5409:0.102004 5482:0.0463599
5533:0.128739 5584:0.0649204 5768:0.0453409 5958:0.0660928 6000:0.114081
6018:0.093675 6076:0.0568362 6078:0.0815319 6121:0.0691536 6182:0.0887539
6219:0.0775283 6235:0.0703129 6449:0.0394806 6450:0.0213164 6578:0.0620504
6584:0.0468761 6605:0.100543 6606:0.0797335 6661:0.0689216 6886:0.0867303
6995:0.180148 7081:0.0835362 7118:0.0849164 7149:0.0846315 7319:0.0451322
7415:0.0417465 7426:0.106298 7428:0.118441 7461:0.0664531 7499:0.0450435
7524:0.0382576 7573:0.0571635 7611:0.0608469 7717:0.109492 7743:0.0877129
7762:0.144791 8006:0.0991949 8027:0.0387654 8032:0.059585 8052:0.0990004
8122:0.0895154

3 -1 171:0.0809199 723:0.302108 1082:0.218679 1261:0.130872 1317:0.167321
1604:0.210675 1626:0.0951768 1827:0.0605716 1837:0.0754022 1999:0.134626
2677:0.152817 2838:0.10718 3135:0.142544 3160:0.349835 3259:0.0371005 3346:0.127322
```

The first document is not related to the "money-fx" category (label = -1). The value (weight) for the first variable (V1) is 0, V2 = 0, ..., V116 = 0.0986899, etc. The second document is related to the target category (label = 1), with V1 = 0, V2 = 0, ..., V21 = 0.0870518, V13 = 0, etc. [The size of the data file is 6 701 KB.](#)

We have transformed the dataset into the attribute-value format. Here are the first rows:

Line	Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10	Col 11	Col 12	Col 13	Col 14	Col 15	Col 16	Col 17	Col 18	Col 19	Col 20	Col 21	Col 22	Col 23	Col 24	Col 25	Col 26	Col 27	Col 28	Col 29	Col 30	Col 31	Col 32	Col 33	Col 34	Col 35	Col 36	Col 37	Col 38	Col 39	Col 40	Col 41	Col 42	Col 43	Col 44	Col 45	Col 46	Col 47	Col 48	Col 49	Col 50	Col 51	Col 52	Col 53	Col 54	Col 55	Col 56	Col 57	Col 58	Col 59	Col 60	Col 61	Col 62	Col 63	Col 64	Col 65	Col 66	Col 67	Col 68	Col 69	Col 70	Col 71	Col 72	Col 73	Col 74	Col 75	Col 76	Col 77	Col 78	Col 79	Col 80	Col 81	Col 82	Col 83	Col 84	Col 85	Col 86	Col 87	Col 88	Col 89	Col 90	Col 91	Col 92	Col 93	Col 94	Col 95	Col 96	Col 97	Col 98	Col 99	Col 100	Col 101	Col 102	Col 103	Col 104	Col 105	Col 106	Col 107	Col 108	Col 109	Col 110	Col 111	Col 112	Col 113	Col 114	Col 115	Col 116	Col 117	Col 118	Col 119	Col 120	Col 121	Col 122	Col 123	Col 124	Col 125	Col 126	Col 127	Col 128	Col 129	Col 130	Col 131	Col 132	Col 133	Col 134	Col 135	Col 136	Col 137	Col 138	Col 139	Col 140	Col 141	Col 142	Col 143	Col 144	Col 145	Col 146	Col 147	Col 148	Col 149	Col 150	Col 151	Col 152	Col 153	Col 154	Col 155	Col 156	Col 157	Col 158	Col 159	Col 160	Col 161	Col 162	Col 163	Col 164	Col 165	Col 166	Col 167	Col 168	Col 169	Col 170	Col 171	Col 172	Col 173	Col 174	Col 175	Col 176	Col 177	Col 178	Col 179	Col 180	Col 181	Col 182	Col 183	Col 184	Col 185	Col 186	Col 187	Col 188	Col 189	Col 190	Col 191	Col 192	Col 193	Col 194	Col 195	Col 196	Col 197	Col 198	Col 199	Col 200	Col 201	Col 202	Col 203	Col 204	Col 205	Col 206	Col 207	Col 208	Col 209	Col 210	Col 211	Col 212	Col 213	Col 214	Col 215	Col 216	Col 217	Col 218	Col 219	Col 220	Col 221	Col 222	Col 223	Col 224	Col 225	Col 226	Col 227	Col 228	Col 229	Col 230	Col 231	Col 232	Col 233	Col 234	Col 235	Col 236	Col 237	Col 238	Col 239	Col 240	Col 241	Col 242	Col 243	Col 244	Col 245	Col 246	Col 247	Col 248	Col 249	Col 250	Col 251	Col 252	Col 253	Col 254	Col 255	Col 256	Col 257	Col 258	Col 259	Col 260	Col 261	Col 262	Col 263	Col 264	Col 265	Col 266	Col 267	Col 268	Col 269	Col 270	Col 271	Col 272	Col 273	Col 274	Col 275	Col 276	Col 277	Col 278	Col 279	Col 280	Col 281	Col 282	Col 283	Col 284	Col 285	Col 286	Col 287	Col 288	Col 289	Col 290	Col 291	Col 292	Col 293	Col 294	Col 295	Col 296	Col 297	Col 298	Col 299	Col 300	Col 301	Col 302	Col 303	Col 304	Col 305	Col 306	Col 307	Col 308	Col 309	Col 310	Col 311	Col 312	Col 313	Col 314	Col 315	Col 316	Col 317	Col 318	Col 319	Col 320	Col 321	Col 322	Col 323	Col 324	Col 325	Col 326	Col 327	Col 328	Col 329	Col 330	Col 331	Col 332	Col 333	Col 334	Col 335	Col 336	Col 337	Col 338	Col 339	Col 340	Col 341	Col 342	Col 343	Col 344	Col 345	Col 346	Col 347	Col 348	Col 349	Col 350	Col 351	Col 352	Col 353	Col 354	Col 355	Col 356	Col 357	Col 358	Col 359	Col 360	Col 361	Col 362	Col 363	Col 364	Col 365	Col 366	Col 367	Col 368	Col 369	Col 370	Col 371	Col 372	Col 373	Col 374	Col 375	Col 376	Col 377	Col 378	Col 379	Col 380	Col 381	Col 382	Col 383	Col 384	Col 385	Col 386	Col 387	Col 388	Col 389	Col 390	Col 391	Col 392	Col 393	Col 394	Col 395	Col 396	Col 397	Col 398	Col 399	Col 400	Col 401	Col 402	Col 403	Col 404	Col 405	Col 406	Col 407	Col 408	Col 409	Col 410	Col 411	Col 412	Col 413	Col 414	Col 415	Col 416	Col 417	Col 418	Col 419	Col 420	Col 421	Col 422	Col 423	Col 424	Col 425	Col 426	Col 427	Col 428	Col 429	Col 430	Col 431	Col 432	Col 433	Col 434	Col 435	Col 436	Col 437	Col 438	Col 439	Col 440	Col 441	Col 442	Col 443	Col 444	Col 445	Col 446	Col 447	Col 448	Col 449	Col 450	Col 451	Col 452	Col 453	Col 454	Col 455	Col 456	Col 457	Col 458	Col 459	Col 460	Col 461	Col 462	Col 463	Col 464	Col 465	Col 466	Col 467	Col 468	Col 469	Col 470	Col 471	Col 472	Col 473	Col 474	Col 475	Col 476	Col 477	Col 478	Col 479	Col 480	Col 481	Col 482	Col 483	Col 484	Col 485	Col 486	Col 487	Col 488	Col 489	Col 490	Col 491	Col 492	Col 493	Col 494	Col 495	Col 496	Col 497	Col 498	Col 499	Col 500	Col 501	Col 502	Col 503	Col 504	Col 505	Col 506	Col 507	Col 508	Col 509	Col 510	Col 511	Col 512	Col 513	Col 514	Col 515	Col 516	Col 517	Col 518	Col 519	Col 520	Col 521	Col 522	Col 523	Col 524	Col 525	Col 526	Col 527	Col 528	Col 529	Col 530	Col 531	Col 532	Col 533	Col 534	Col 535	Col 536	Col 537	Col 538	Col 539	Col 540	Col 541	Col 542	Col 543	Col 544	Col 545	Col 546	Col 547	Col 548	Col 549	Col 550	Col 551	Col 552	Col 553	Col 554	Col 555	Col 556	Col 557	Col 558	Col 559	Col 560	Col 561	Col 562	Col 563	Col 564	Col 565	Col 566	Col 567	Col 568	Col 569	Col 570	Col 571	Col 572	Col 573	Col 574	Col 575	Col 576	Col 577	Col 578	Col 579	Col 580	Col 581	Col 582	Col 583	Col 584	Col 585	Col 586	Col 587	Col 588	Col 589	Col 590	Col 591	Col 592	Col 593	Col 594	Col 595	Col 596	Col 597	Col 598	Col 599	Col 600	Col 601	Col 602	Col 603	Col 604	Col 605	Col 606	Col 607	Col 608	Col 609	Col 610	Col 611	Col 612	Col 613	Col 614	Col 615	Col 616	Col 617	Col 618	Col 619	Col 620	Col 621	Col 622	Col 623	Col 624	Col 625	Col 626	Col 627	Col 628	Col 629	Col 630	Col 631	Col 632	Col 633	Col 634	Col 635	Col 636	Col 637	Col 638	Col 639	Col 640	Col 641	Col 642	Col 643	Col 644	Col 645	Col 646	Col 647	Col 648	Col 649	Col 650	Col 651	Col 652	Col 653	Col 654	Col 655	Col 656	Col 657	Col 658	Col 659	Col 660	Col 661	Col 662	Col 663	Col 664	Col 665	Col 666	Col 667	Col 668	Col 669	Col 670	Col 671	Col 672	Col 673	Col 674	Col 675	Col 676	Col 677	Col 678	Col 679	Col 680	Col 681	Col 682	Col 683	Col 684	Col 685	Col 686	Col 687	Col 688	Col 689	Col 690	Col 691	Col 692	Col 693	Col 694	Col 695	Col 696	Col 697	Col 698	Col 699	Col 700	Col 701	Col 702	Col 703	Col 704	Col 705	Col 706	Col 707	Col 708	Col 709	Col 710	Col 711	Col 712	Col 713	Col 714	Col 715	Col 716	Col 717	Col 718	Col 719	Col 720	Col 721	Col 722	Col 723	Col 724	Col 725	Col 726	Col 727	Col 728	Col 729	Col 730	Col 731	Col 732	Col 733	Col 734	Col 735	Col 736	Col 737	Col 738	Col 739	Col 740	Col 741	Col 742	Col 743	Col 744	Col 745	Col 746	Col 747	Col 748	Col 749	Col 750	Col 751	Col 752	Col 753	Col 754	Col 755	Col 756	Col 757	Col 758	Col 759	Col 760	Col 761	Col 762	Col 763	Col 764	Col 765	Col 766	Col 767	Col 768	Col 769	Col 770	Col 771	Col 772	Col 773	Col 774	Col 775	Col 776	Col 777	Col 778	Col 779	Col 780	Col 781	Col 782	Col 783	Col 784	Col 785	Col 786	Col 787	Col 788	Col 789	Col 790	Col 791	Col 792	Col 793	Col 794	Col 795	Col 796	Col 797	Col 798	Col 799	Col 800	Col 801	Col 802	Col 803	Col 804	Col 805	Col 806	Col 807	Col 808	Col 809	Col 810	Col 811	Col 812	Col 813	Col 814	Col 815	Col 816	Col 817	Col 818	Col 819	Col 820	Col 821	Col 822	Col 823	Col 824	Col 825	Col 826	Col 827	Col 828	Col 829	Col 830	Col 831	Col 832	Col 833	Col 834	Col 835	Col 836	Col 837	Col 838	Col 839	Col 840	Col 841	Col 842	Col 843	Col 844	Col 845	Col 846	Col 847	Col 848	Col 849	Col 850	Col 851	Col 852	Col 853	Col 854	Col 855	Col 856	Col 857	Col 858	Col 859	Col 860	Col 861	Col 862	Col 863	Col 864	Col 865	Col 866	Col 867	Col 868	Col 869	Col 870	Col 871	Col 872	Col 873	Col 874	Col 875	Col 876	Col 877	Col 878	Col 879	Col 880	Col 881	Col 882	Col 883	Col 884	Col 885	Col 886	Col 887	Col 888	Col 889	Col 890	Col 891	Col 892	Col 893	Col 894	Col 895	Col 896	Col 897	Col 898	Col 899	Col 900	Col 901	Col 902	Col 903	Col 904	Col 905	Col 906	Col 907	Col 908	Col 909	Col 910	Col 911	Col 912	Col 913	Col 914	Col 915	Col 916	Col 917	Col 918	Col 919	Col 920	Col 921	Col 922	Col 923	Col 924	Col 925	Col 926	Col 927	Col 928	Col 929	Col 930	Col 931	Col 932	Col 933	Col 934	Col 935	Col 936	Col 937	Col 938	Col 939	Col 940	Col 941	Col 942	Col 943	Col 944	Col 945	Col 946	Col 947	Col 948	Col 949	Col 950	Col 951	Col 952	Col 953	Col 954	Col 955	Col 956	Col 957	Col 958	Col 959	Col 960	Col 961	Col 962	Col 963	Col 964	Col 965	Col 966	Col 967	Col 968	Col 969	Col 970	Col 971	Col 972	Col 973	Col 974	Col 975	Col 976	Col 977	Col 978	Col 979	Col 980	Col 981	Col 982	Col 983	Col 984	Col 985	Col 986	Col 987	Col 988	Col 989	Col 990	Col 991	Col 992	Col 993	Col 994	Col 995	Col 996	Col 997	Col 998	Col 999	Col 1000
------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	----------

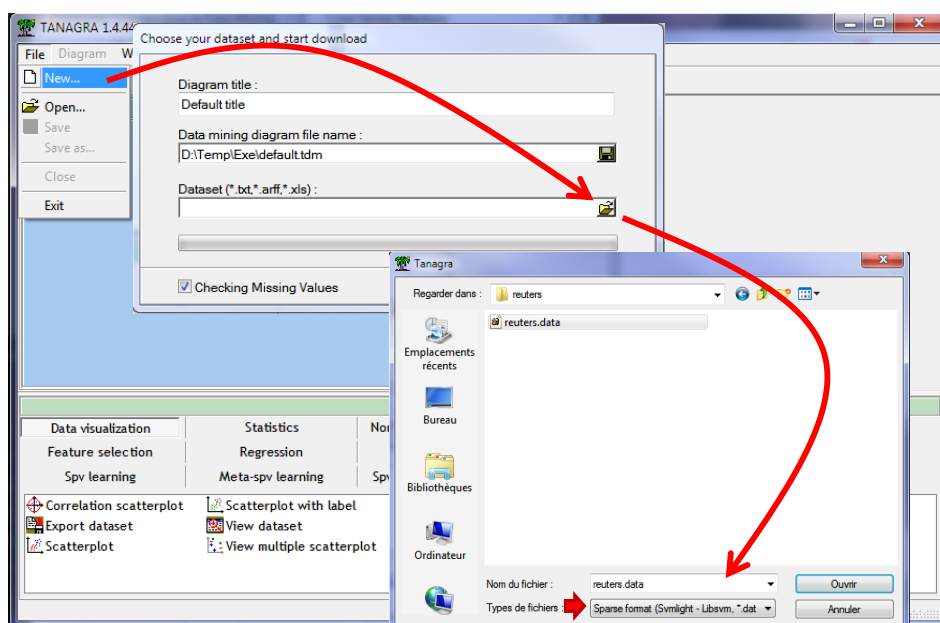
The size of the data file is now 183 309 KB. It is 27 times more sizeable!

3 Processing the sparse format with Tanagra

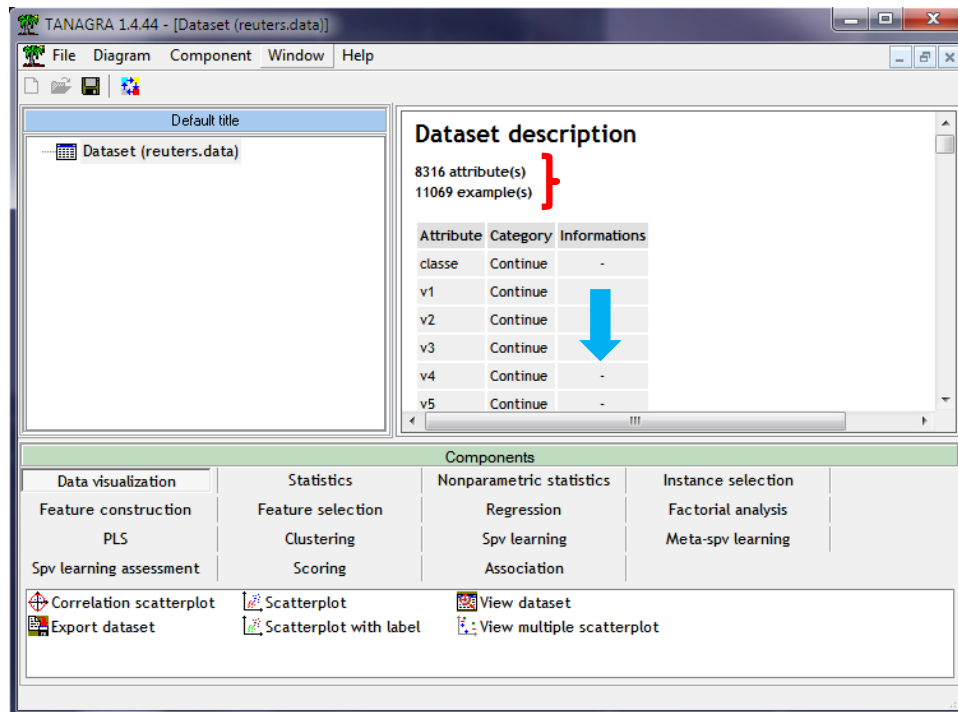
Starting from the 1.4.44 version, Tanagra can handle a sparse data file format. The format is based on the one provided by the SvmLight, Libsvm, Libsvm libraries. The filename extension must be ".dat" or ".data" for that Tanagra could identify it.

In the following, we show how to import this kind of data file, to make the needed transformations for classification task, to build the predictive model on the learning sample, and to evaluate the quality of the model using a ROC curve computed on the test sample.

3.1 Data file importation

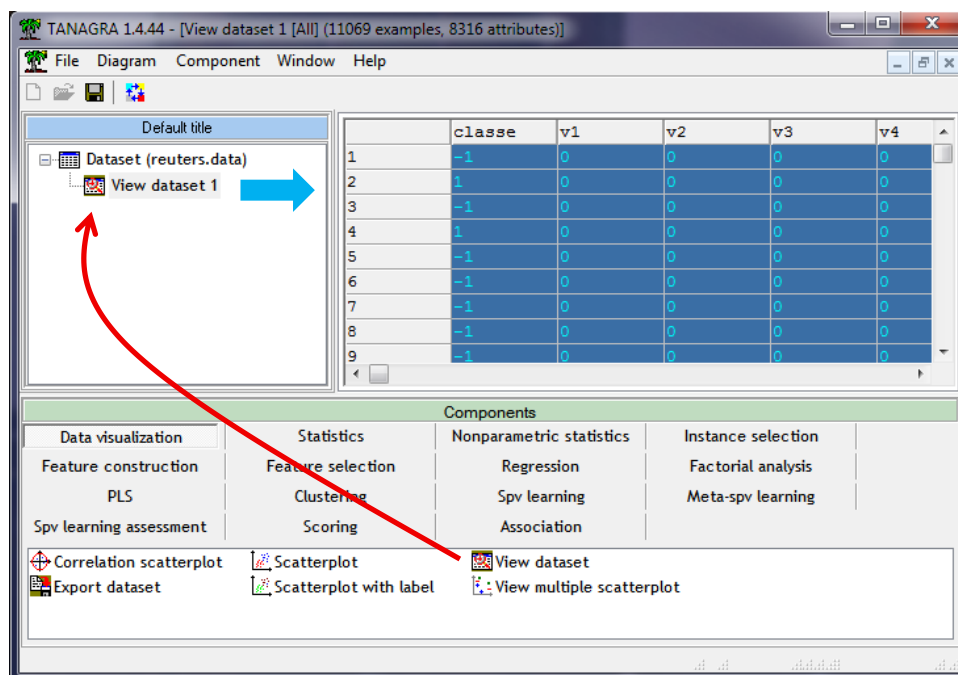


After launching Tanagra, we activate the menu FILE / NEW to create a new project and import the database. We select the "reuters.data" file. We have 11069 instances and 8316 columns. The names of the variables are assigned automatically ("classe" for the target attribute, then "V1", "V2", etc. for the following ones).



3.2 Visualization

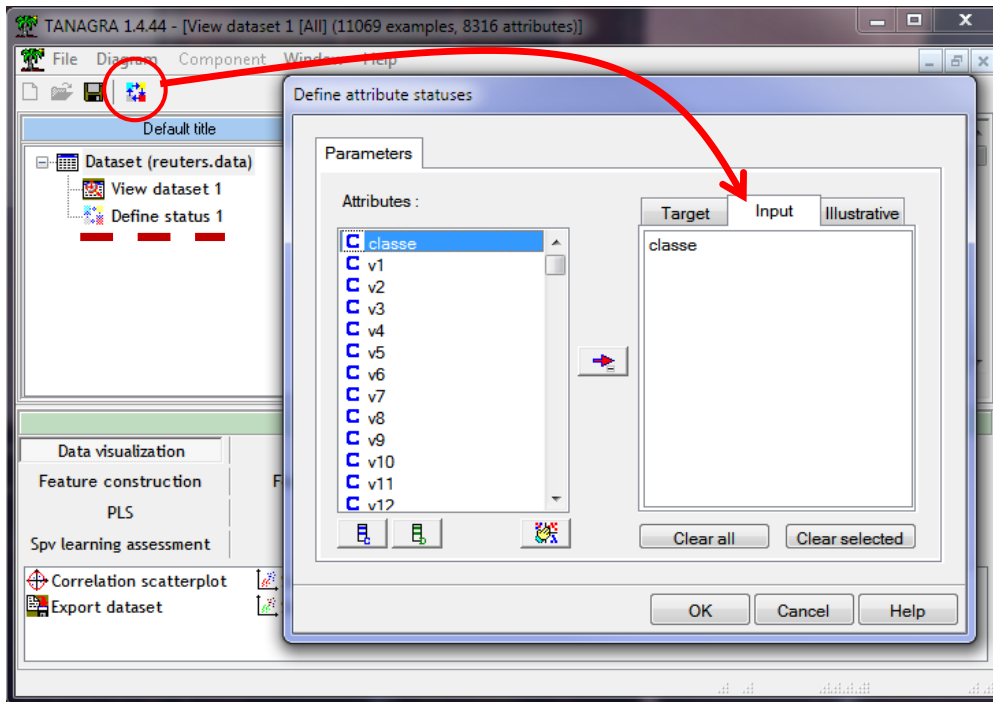
To check if the importation process was done properly, we visualize the dataset using the VIEW DATASET component (DATA VISUALIZATION tab).



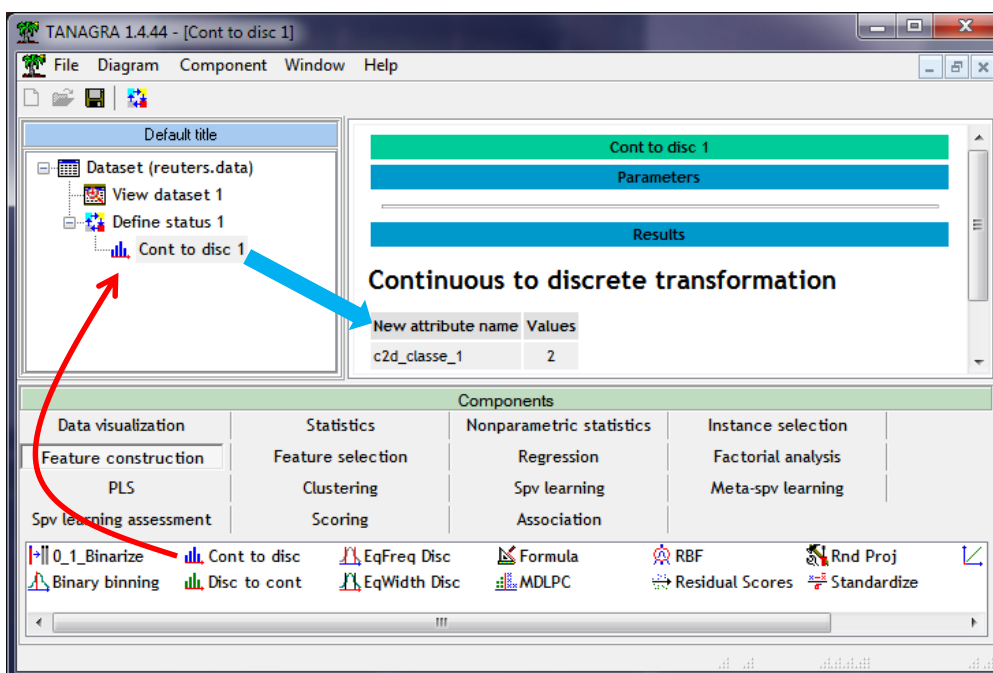
The file is in "sparse" format. But the data is encoded internally in the classic attribute-value format. Memory usage can quickly become important. It is 594,204 KB at this step. This solution is not optimal, but it allows us to process the sparse and dense data in the same way.

3.3 Recoding the target attribute

Because the "classe" variable is numeric, we cannot use it directly for the classification process. We must recode it. For that, we insert the DEFINE STATUS component into the diagram and we set the "classe" variable as INPUT.



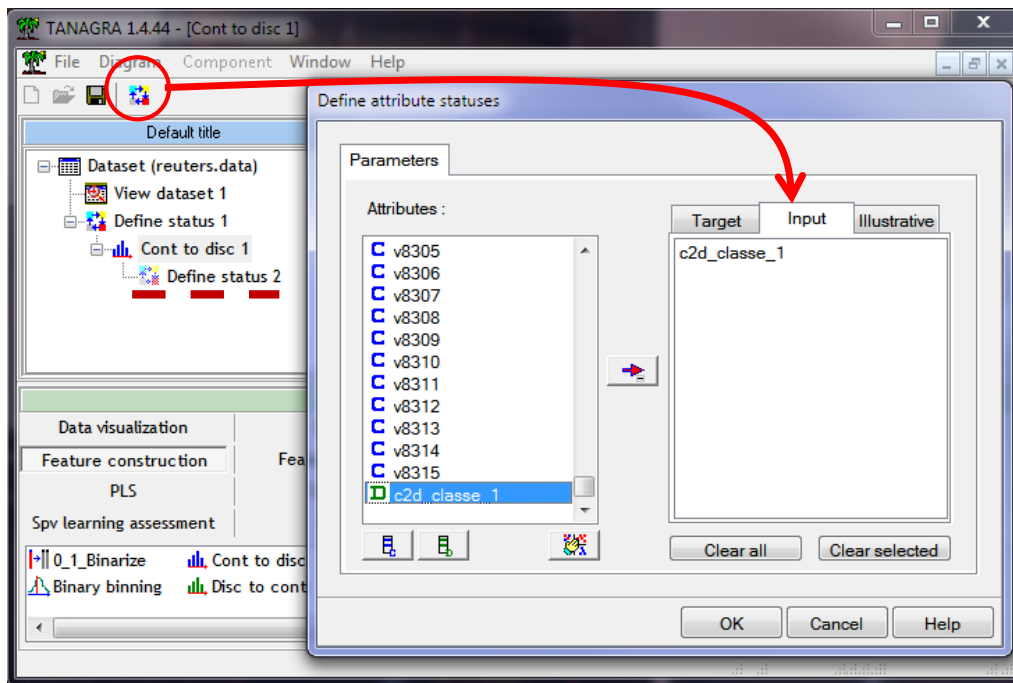
Then we add the CONT TO DISC (FEATURE CONSTRUCTION tab) component. It recodes the variable by assigning a category for each distinct value.



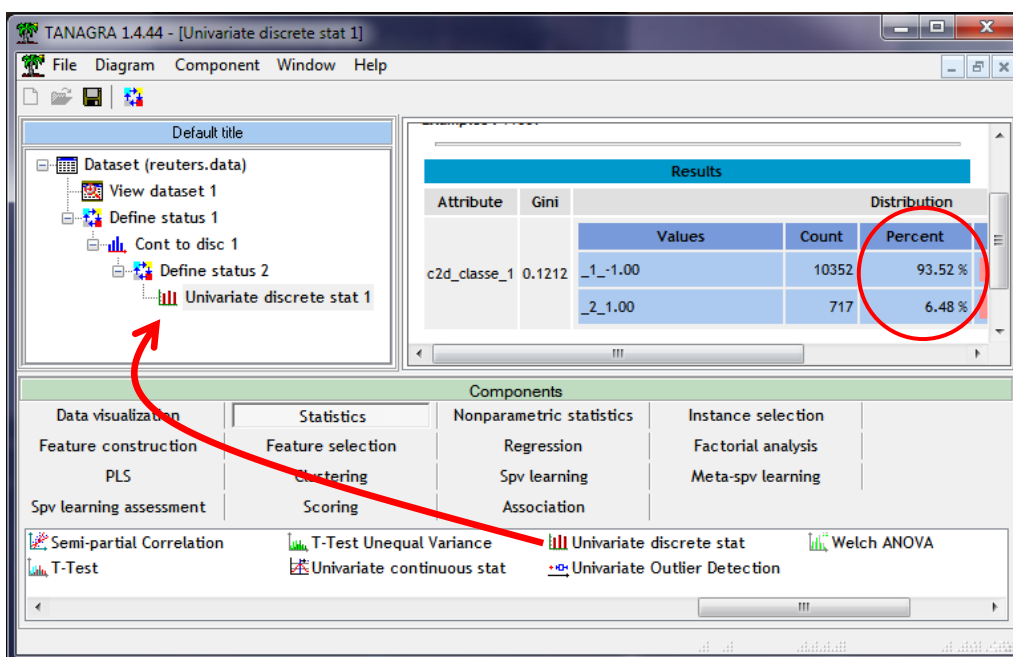
After we click on the contextual menu VIEW, we observe that the new variable C2D_CLASSE_1 is categorical with 2 values {"-1", "1"}.

3.4 Classes distribution

We can compute the classes distribution. We insert the DEFINE STATUS into the diagram. We set the new attribute C2D_CLASSE_1 as INPUT.



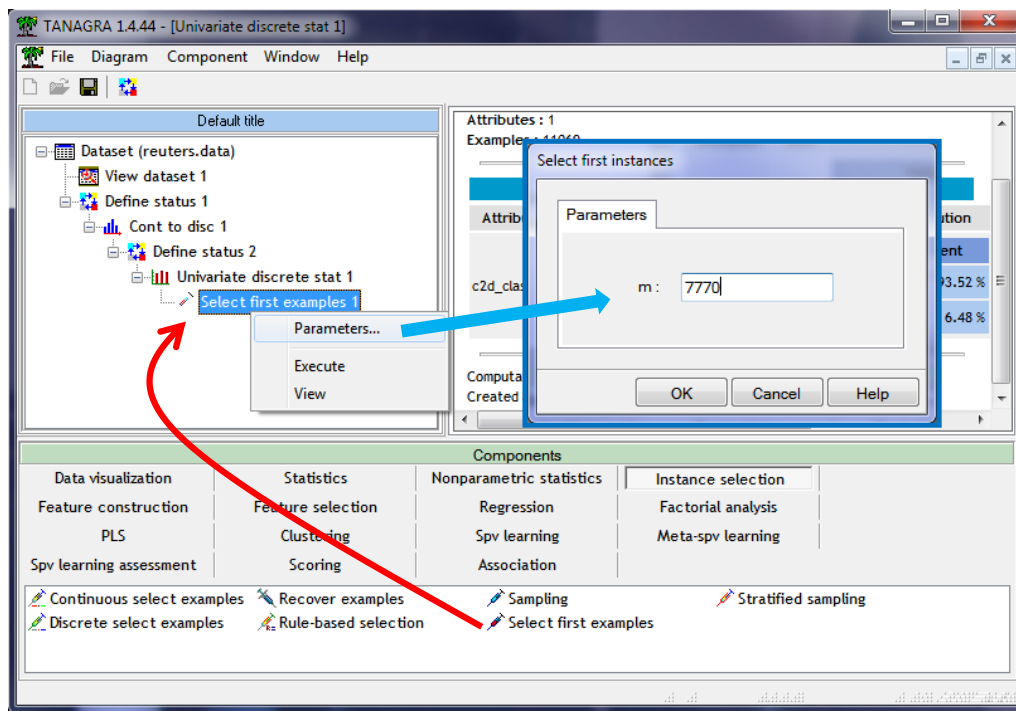
Then, we add the UNIVARIATE DISCRETE STAT (STATISTICS tab) component.



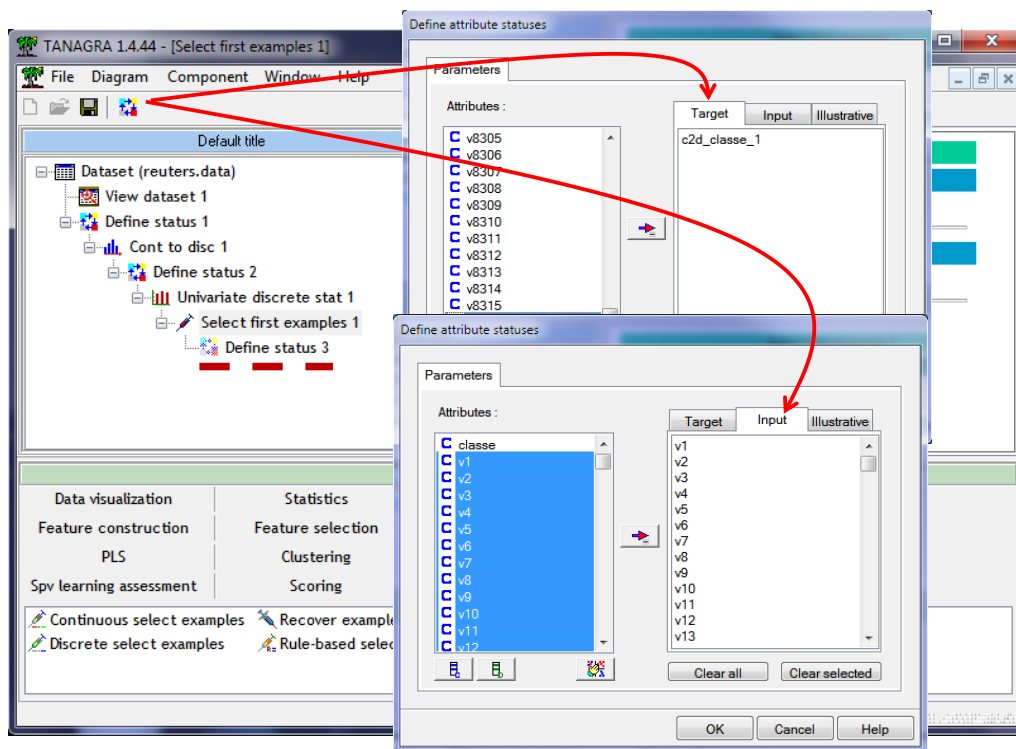
We observe that the classes are rather imbalanced (6.48% of instances for the target value "1").

3.5 Partitioning the dataset

The learning and testing sets were merged before the data importation. We must specify the two samples into Tanagra. We insert the SELECT FIRST EXAMPLES component (INSTANCE SELECTION tab). The 7770 first instances correspond to the learning set.

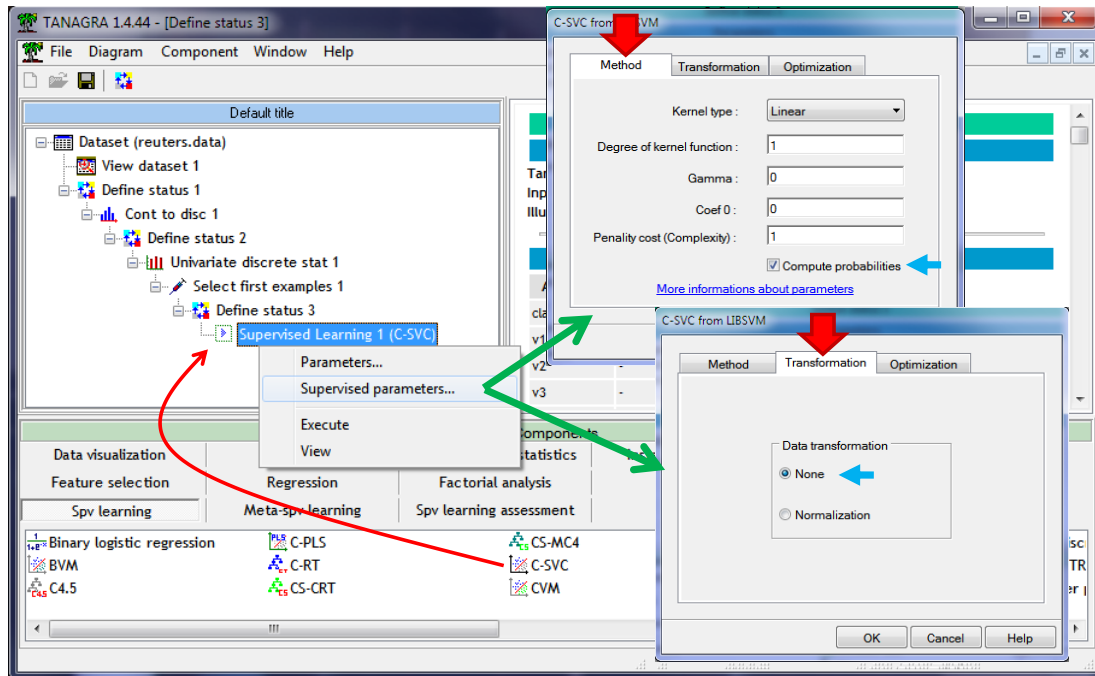


3.6 Learning process - C-SVC from the Libsvm library



We want to model the relationship between the class and the description of the documents. To specify the role of the variables, we add the DEFINE STATUS into the diagram. We set C2D_CLASSE_1 as TARGET, (V1, ..., V8315) as INPUT. Beware, the CLASS column should no longer be used at this stage (and subsequently) of our study.

We add the C-SVC (SPV LEARNING tab) component. It implements a Linear SVM (support vector machine). We set the following parameters:



Two parameters are essential here: we ask the calculation of the class assignment probabilities needed for the construction of the ROC curve; it is not necessary to normalize the variables.

We click on the VIEW contextual menu to launch the calculation. The resubstitution error rate is 0.59%. This value is not relevant here because the classes are imbalanced.

Classifier performances

Error rate		0.0059				
Values prediction		Confusion matrix				
Value	Recall	1-Precision				
			1-1.00	_2_-1.00	Sum	
1-1.00	0.9965	0.0029	_1_-1.00	7207	25	7232
2-1.00	0.9610	0.0461	_2_-1.00	21	517	538
			Sum	7228	542	7770

Classifier characteristics

Data description

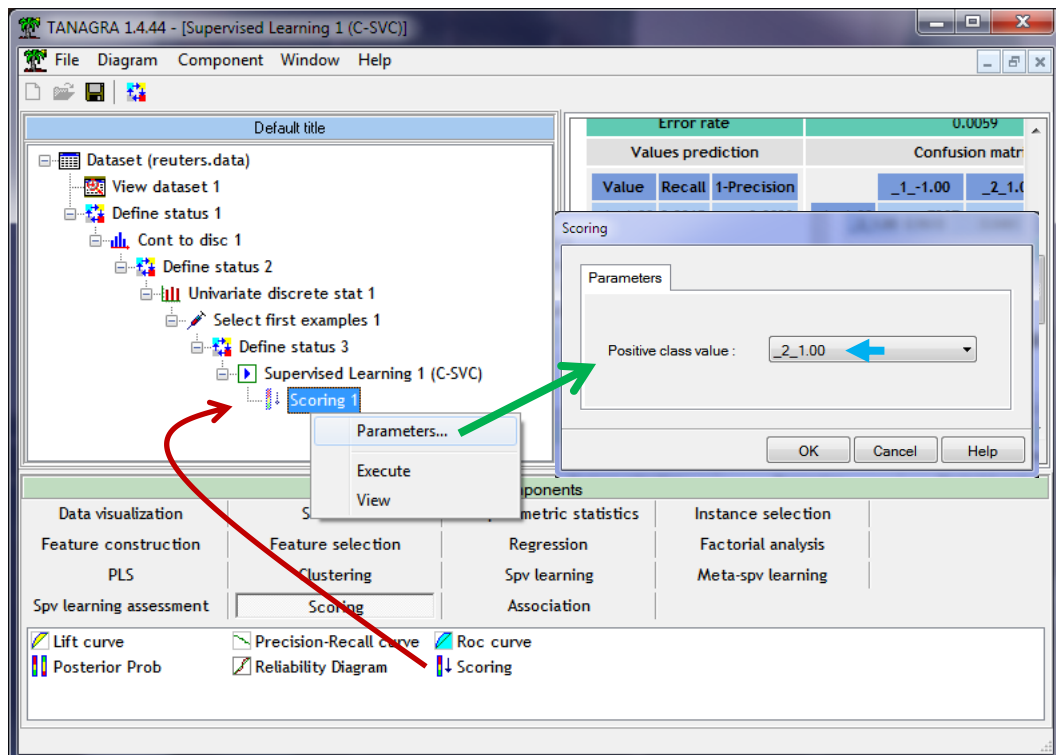
Target attribute	c2d_classe_1 (2 values)
# descriptors	8315

SVM characteristics

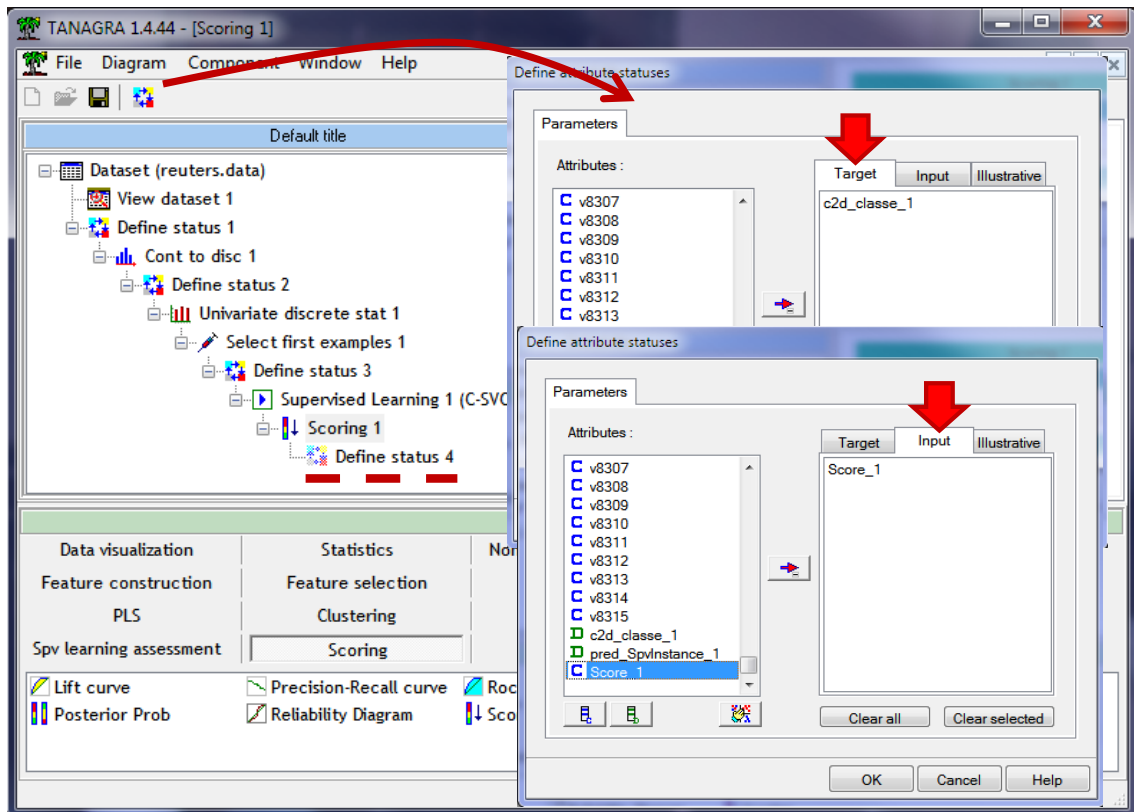
Characteristic	Value
# classes	2
# support vectors	789
# support vectors for each class	
# sv. for _1_-1.00	519
# sv. for _2_-1.00	270

3.7 Scoring and construction of the ROC curve on the test set

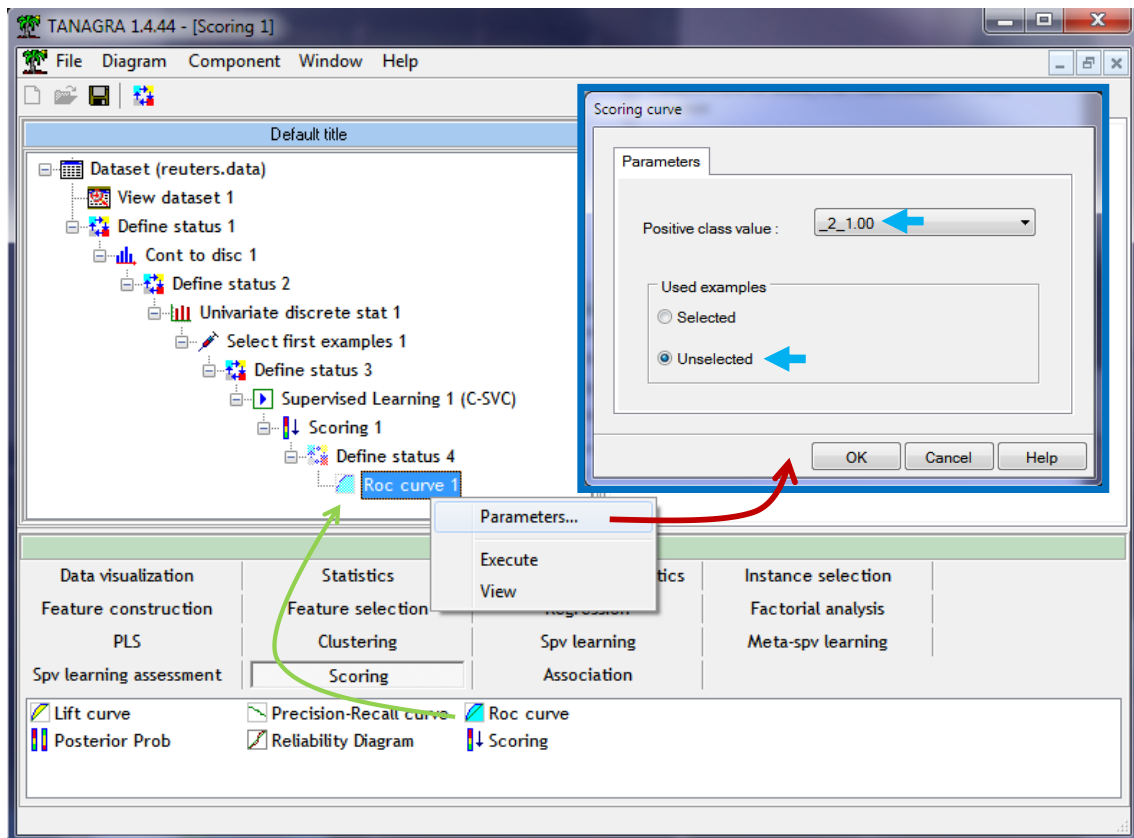
To construct the ROC curve, we must calculate the probability of belonging to the positive (target) modality "1" of the class attribute on the test set (Tanagra calculates the scores for the whole dataset, including the test set). We insert the SCORING component (SCORING tab). We want to calculate the scores for the modality "1" of the target attribute.



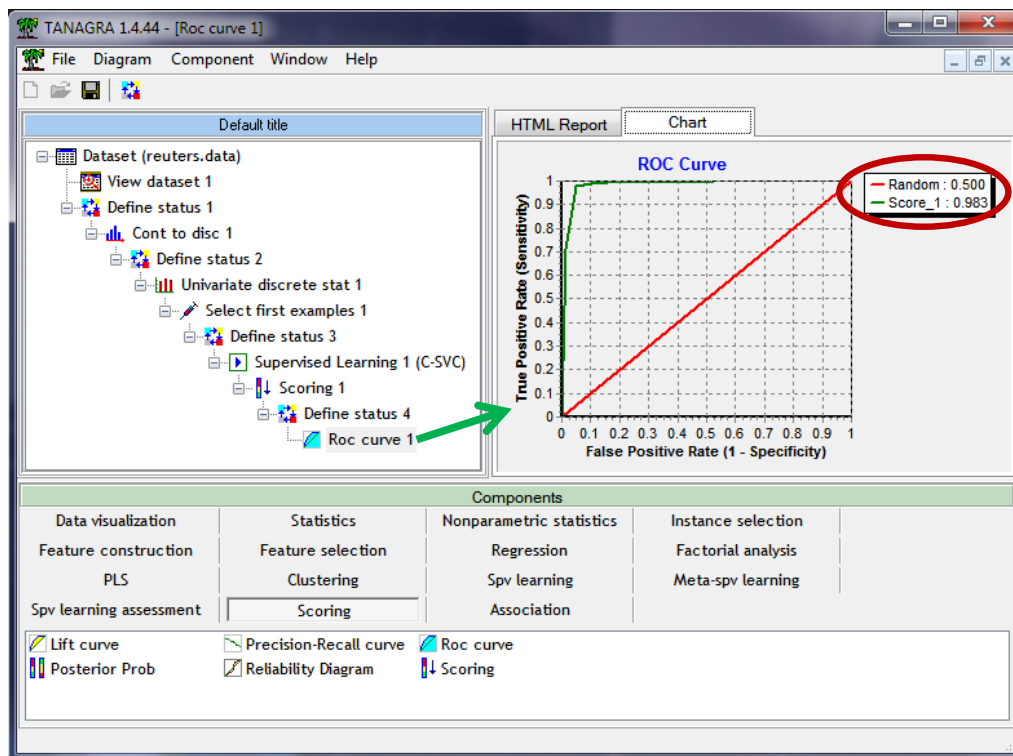
Then, we add the DEFINE STATUS component: C2D_CLASSE_1 is the TARGET; SCORE_1, computed previously, is the INPUT attribute.



We add the ROC CURVE (SCORING tab) component. We specify the parameters so that the curve is constructed for the modality "1" of the target attribute (POSITIVE CLASS VALUE = 1), on the unselected instances i.e. the test set (USED EXAMPLES = UNSELECTED).



We see that our model is efficient. The area under curve (**AUC**) is equal to **0.983** i.e. a randomly chosen positive instance (of the class "1") has 98.3% chances of having a higher score than a negative instance (of the class "-1").

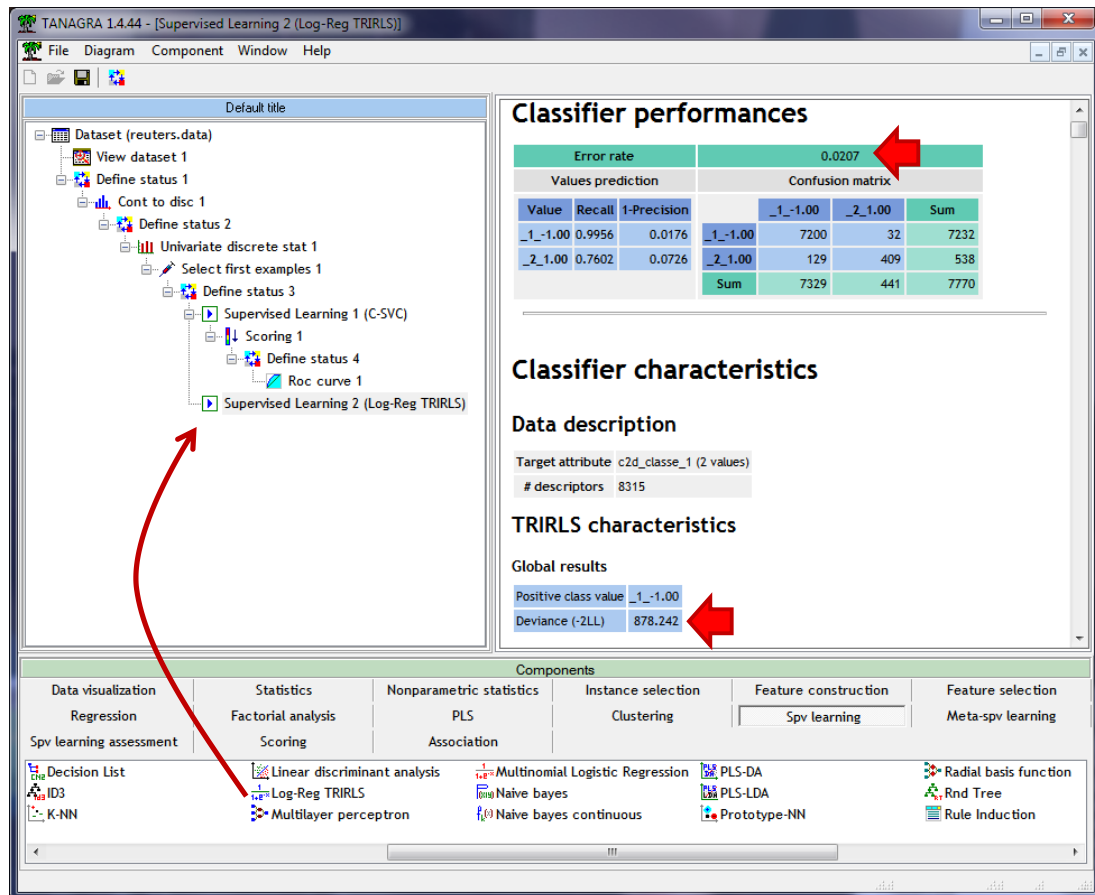


3.8 Comparison with logistic regression

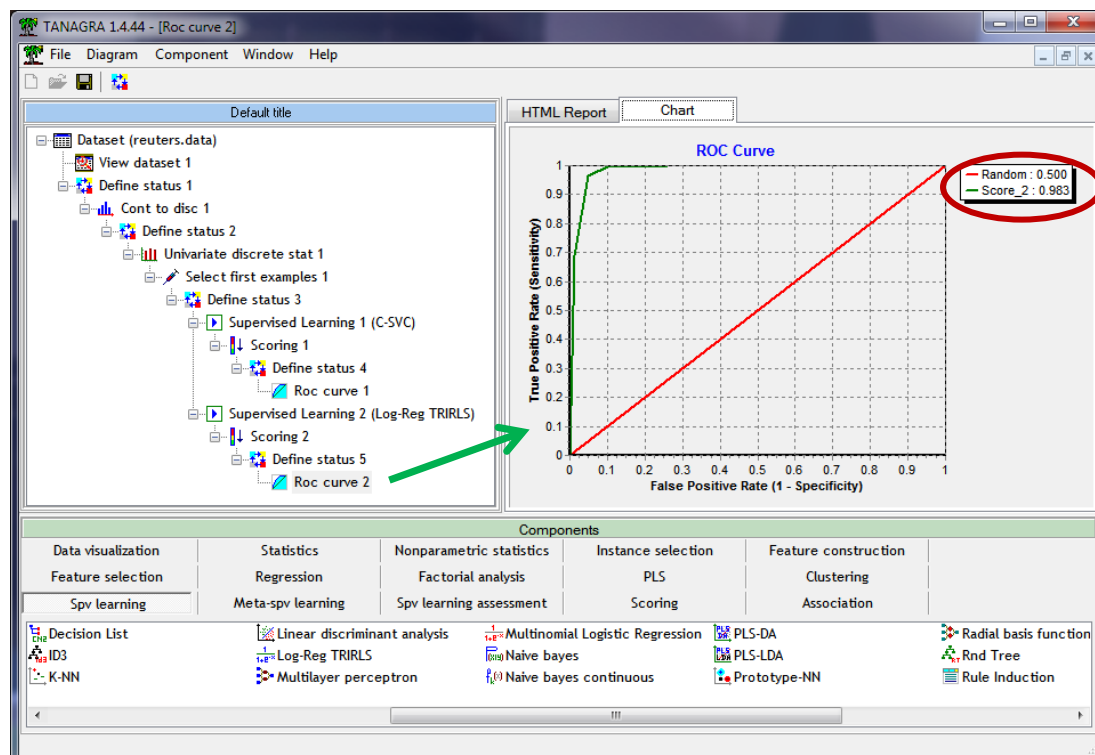
We want to lead the same analysis but using logistic regression. We use an implementation of TRIRLS approach (iterative reweighted least squares algorithm for logistic regression⁴). We added the AUTONLAB library as a DLL in the Tanagra (from the 1.4.44 version) (<http://autonlab.org/autonweb/10538>). If the library seems not really efficient on standard dataset (few variables, large number of instances), it seems much better on wide dataset (large number of descriptors).

We insert the LOG-REG TRIRLS (SPV LEARNING tab) component into the diagram. We obtain the model deviance ($-2LL = 878.242$) and the coefficients of the regression equation. The resubstitution error rate is 2.07%. But we know that this criterion is not relevant in our context.

⁴ https://en.wikipedia.org/wiki/Iteratively_reweighted_least_squares



More relevant are the ROC curve and the AUC criterion (computed on the test set = 0.983).



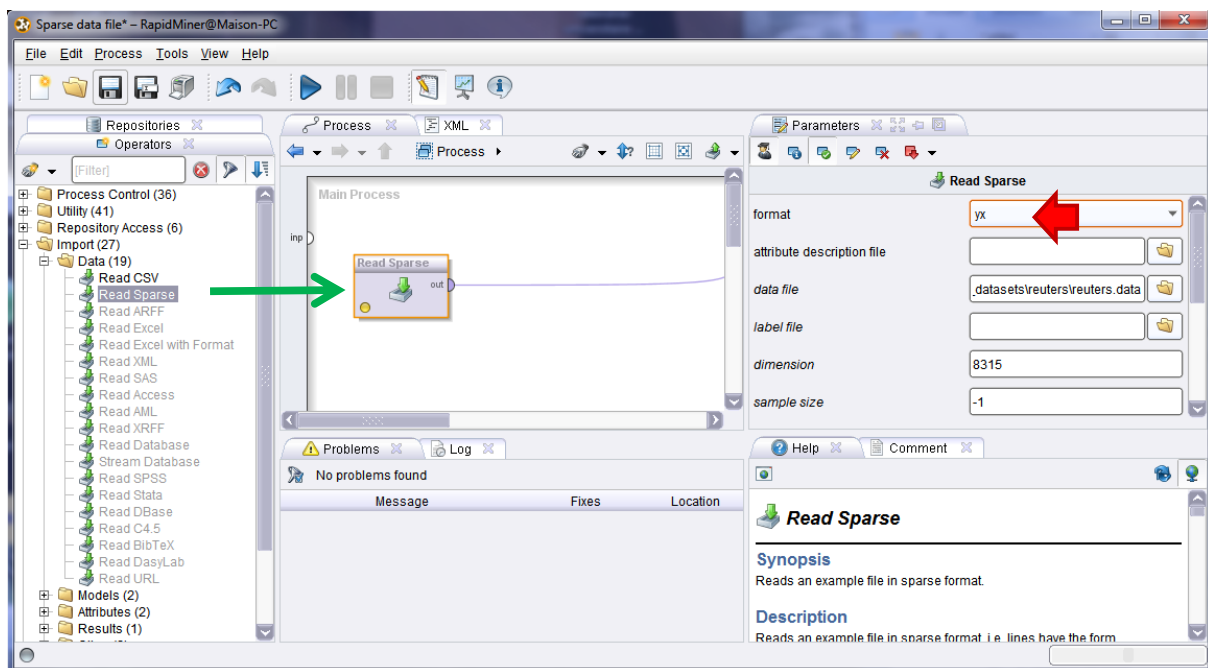
The logistic regression is as well as the linear SVM for our text categorization problem.

4 Sparse data file processing with other tools

Of course, other data mining tools can handle the sparse data file format (Svmlight, Libsvm). We describe briefly in this section the functionalities of RapidMiner and Weka.

4.1 RapidMiner

RapidMiner 5.2.006 (<http://rapid-i.com/content/view/181/190/>) can read the sparse files as we describe them in this tutorial. Better even, we can specify precisely the organization of the data by means of various parameters. The FORMAT option is essential. It allows to specify the presence of the label into the file (in the case of clustering, it is not needed) and, possibly, its position. For our dataset, we set FORMAT = YX because the label is in the first position in the row. The other settings are described in the contextual help.



RapidMiner provides a statistical summary of the variables after loading the data file (min, max, mean, standard deviation). It should also be noted that the names of the variables are automatically assigned.

For comparison, I show the indicators computed with the UNIVARIATE CONTINUOUS STAT component (STATISTICS tab) of Tanagra. We have of course the same results.

TANAGRA Univariate Continuous Stat

Attribute	Min	Max	Average	Std-dev	Std-dev/avg
classe	-1	1	-0.8704	0.4923	-0.5655
v1	0	0.518098	0.0003	0.0100	33.0697
v2	0	0.236209	0.0001	0.0050	38.9161
v3	0	0.240174	0.0001	0.0034	61.8230
v4	0	0.214651	0.0001	0.0035	60.7377
v5	0	0.584598	0.0001	0.0074	55.7416
v6	0	0.705593	0.0002	0.0097	61.4414
v7	0	0.566478	0.0012	0.0176	14.6031
v8	0	0.285344	0.0005	0.0084	15.9816
v9	0	0.263867	0.0001	0.0044	42.7770
v10	0	0.197253	0.0000	0.0026	62.4435
v11	0	0.773622	0.0001	0.0082	72.6606
v12	0	0.737673	0.0002	0.0108	61.3924
v13	0	0.257434	0.0001	0.0033	63.6812
v14	0	0.29251	0.0002	0.0063	28.3777

4.2 Weka

Weka 3.7.5 (<http://www.cs.waikato.ac.nz/ml/weka/>) knows also to read the Svmight or Libsvm sparse format. The filename extension must be “.libsvm”.

Weka Explorer

Ouvrir

Rechercher dans : reuters

reuters.libsvm

Nom du fichier : reuters.libsvm

Fichiers de type : libsvm data files (*.libsvm)

Ouvrir

Annuler

5 Conclusion

The "sparse" data file format enables to reduce the file size by adopting a customized data representation. Like any compression algorithm, there are contexts where it generates zero or even negative gains. When we are faced with a standard database (called also "dense database"), with a very small proportion of zero values, it is not efficient. On the other hand, it becomes relevant when we handle pre-processed data from a collection of unstructured data, where the descriptors are generated automatically with a large proportion of zero values. We have taken the example of text mining context in this tutorial.