Laboratoire 3 à 6

Par Olivier Arteau et Simon Grondin

Arbre de décision

Arbre	Ensemble d'apprentissage	Élagage (facteur)	Taille	Taux d'erreur	VP	VN	FP	FN
A1	spambase-train	N/A	2300	6.00%	816	1347	66	72
A2	spambase-train	0.25	2300	6.21%	808	1350	63	80
A3	spambase-train-50p	N/A	1150	9.17%	777	1313	100	111
A4	spambase-train-50p	0.25	1150	8.82%	770	1328	85	118
A5	spambase-train-25p	N/A	575	11.73%	686	1345	68	202
A6	spambase-train-25p	0.25	575	11.82%	686	1343	70	202
A7	spambase-train-25p-5b	N/A	575	15.30%	664	1285	128	224
A8	spambase-train-25p-5b	0.25	575	13.08%	700	1300	113	188
A9	spambase-train-25p-10b	N/A	575	17.38%	649	1252	161	239
A10	spambase-train-25p-10b	025	575	16.64%	670	1248	165	218
A11	spambase-train-25p-20b	N/A	575	21.69%	583	1219	194	305
A12	spambase-train-25p-20b	0.25	575	21.08%	669	1147	266	219

Pour l'ensemble d'apprentissage de base, on remarque que l'élagage n'améliore pas la performance. Ceci peut s'expliquer par le fait que les données sont probablement de qualité et que l'apprentissage sans élagage ne fait pas de sur-apprentissage. On observe le même phénomène pour l'ensemble qui contient 50% des données et 25% des données. Pour ces échantillions, les différences de taux d'erreur sont négligables et peuvent être attribués à la variance du choix aléatoire des éléments de l'ensemble d'apprentissage et de test.

Par contre lorsque l'on regarde les échantillions avec du bruit, on voit bien que l'élagage permet d'améliorer la performance de l'apprentissage puisqu'il permet d'éviter d'apprendre le bruit. Pour l'échantillion avec 20% de bruit, la différence de performance entre avec l'élagage et sans l'élagage est plus mitigé. Ceci est probablement dû au fait que l'élagage ne permet d'éliminer complètement le bruit et lorsqu'il y en a trop, l'algorithme n'est just plus performant du tout.

Bayes

Arbre	Ensemble d'apprentissage	Kernel Est.	Super. Discr.	Taille	Taux d'erreur	VP	VN	FP	FN
A1	spambase-train	False	False	2300	20.77	836	987	426	52
A2	spambase-train	True	False	2300	20.03	867	973	440	21
A3	spambase-train	False	True	2300	10.691	727	1328	85	161
A4	spambase-train-50p	False	False	1150	20.99	828	990	423	60
A5	spambase-train-50p	True	False	1150	17.29	854	1049	364	34
A6	spambase-train-50p	False	True	1150	11.56	703	1332	81	185
A7	spambase-train-25p	False	False	575	19.20	836	1023	390	52
A8	spambase-train-25p	True	False	575	16.68	844	1073	340	44
A9	spambase-train-25p	False	True	575	12.29	680	1338	75	208
A10	spambase-train-25p-5b	False	False	575	26.03	833	869	544	55
A11	spambase-train-25p-5b	True	False	575	11.08	778	1268	145	110
A12	spambase-train-25p-5b	False	True	575	13.68	672	1314	99	216
A13	spambase-train-25p-10b	False	False	575	25.46	837	878	535	51
A14	spambase-train-25p-10b	True	False	575	11.47	769	1268	145	119
A15	spambase-train-25p-10b	False	True	575	13.03	683	1318	95	205
A16	spambase-train-25p-20b	False	False	575	16.60	647	1272	141	241
A17	spambase-train-25p-20b	True	False	575	24.64	390	1344	69	498
A18	spambase-train-25p-20b	False	True	575	14.68	675	1288	125	213

L'algorithme de Bayes est excellent pour ce type de problème, spécialement le mode Kernel Estimator lorsqu'il y a un peu de bruit. Autrement, c'est le mode Supervized Discretization qui produit généralement les meilleurs résultats, avec des taux d'erreurs descendant dans les 10% à 15%. Le mode “standard” est en général le moins performant. L'algorithme de Bayes survit bien au bruit.

K plus proche voisin

Arbre	Ensemble d'apprentissage	k	Taille	Taux d'erreur	VP	VN	FP	FN
A1	spambase-train	1	2300	7.56%	802	1325	88	86
A2	spambase-train	2	2300	10.86%	807	1244	169	81
A3	spambase-train	3	2300	11.30%	754	1287	126	134
A4	spambase-train	5	2300	11.12%	750	1295	118	138
A5	spambase-train	10	2300	11.21%	742	1301	112	146
A6	spambase-train-50p	1	1150	12.65%	719	1291	122	169
A7	spambase-train-50p	2	1150	13.47%	761	1230	183	127
A8	spambase-train-50p	3	1150	15.38%	712	1235	178	176
A9	spambase-train-50p	5	1150	13.86%	715	1267	146	173
A10	spambase-train-50p	10	1150	15.17%	683	1269	144	205
A11	spambase-train-25p	1	575	15.38%	685	1262	151	203
A12	spambase-train-25p	2	575	15.78%	755	1183	230	133
A13	spambase-train-25p	3	575	15.30%	675	1274	139	213
A14	spambase-train-25p	5	575	14.86%	659	1300	113	229
A15	spambase-train-25p	10	575	14.21%	664	1310	103	224
A16	spambase-train-25p-5b	1	575	19.17%	675	1185	228	213
A17	spambase-train-25p-5b	2	575	20.12%	760	1078	335	128
A18	spambase-train-25p-5b	3	575	16.17%	673	1256	157	215
A19	spambase-train-25p-5b	5	575	14.86%	658	1301	112	230
A20	spambase-train-25p-5b	10	575	13.95%	677	1303	110	211
A21	spambase-train-25p-10b	1	575	22.64%	652	1128	285	236
A22	spambase-train-25p-10b	2	575	23.25%	754	1012	401	134
A23	spambase-train-25p-10b	3	575	17.99%	659	1228	185	229
A24	spambase-train-25p-10b	5	575	15.69%	660	1280	133	228
A25	spambase-train-25p-10b	10	575	14.30%	691	1281	132	197
A26	spambase-train-25p-20b	1	575	30.07%	616	993	420	272
A27	spambase-train-25p-20b	2	575	32.38%	729	827	586	159
A28	spambase-train-25p-20b	3	575	25.38%	625	1092	321	263
A29	spambase-train-25p-20b	5	575	21.21%	655	1158	255	233
A30	spambase-train-25p-20b	10	575	19.21%	696	1163	250	192

Pour l'ensemble d'apprentissage de base, on remarque que la meilleur performance ce trouve avec le paramètre k=1. Ceci peut s'expliquer par le fait que les données sont de bonnes qualité et que l'absence de bruit fait en sorte que l'élément le plus près est plus souvent le meilleur choix. On remarque le même phénomène avec l'ensemble qui contient la moitié des échantillions. Par contre pour l'échantillion qui contient 25% des données la performance s'améliore plus le k est grand. Comme cet échantillion contient beacoup moins de données, le voisin le plus près n'est pas nécessairement très près et il est alors intéressant de comparer avec plus de voisin pour avoir un meilleur résultat. Pour les échantillions avec du bruit on remarque le même phénomène qu'avec l'échantillion qui contient peu de données. Plus le k est élévé meilleur la performance est. Dans ce cas-ci, cela peut être attribuable au fait que l'on réduit l'impact du bruit lorsque davantage de voisins sont pris en compte pour faire la classification.

Conclusion

Nous remarquons que les algorithmes KNN et les Arbres de Décisions sont excellents lorsque nous avons beaucoup de données et qu'elles sont très propres. Par contre, lorsque nous ajoutons du bruit et réduisons la taille des données, c'est l'algorithme de Bayes, en particulier avec le mode Kernel Estimator qui produit des résultats fiables et constants dans les environs de 13%.