Empirical Study of the Performance of Six Algorithms on Five Binary Classification Problems

Maximilian Siemers   SIEMERSM@GMAIL.COM

Department of Cognitive Science
University of California, San Diego
San Diego, CA 92117, USA

Class: COGS 118A: Introduction to Machine Learning I
Instructor: Prof. Zhuowen Tu

Abstract

Keywords:

1. Introduction

2. Method

3. Experiment (Results)

3.1 Hyperparameters and Performance

3.2 Train/Test split and Performance

3.3 Overall Classifier Performance

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante.
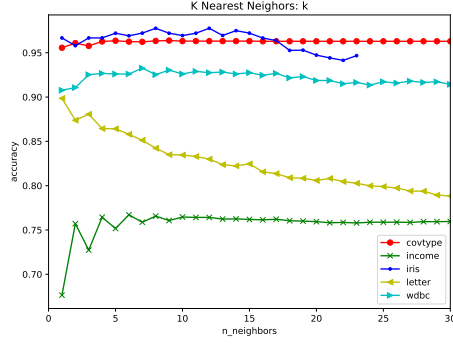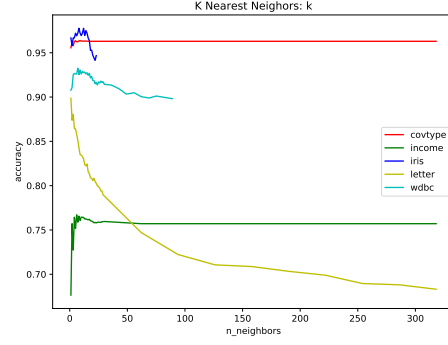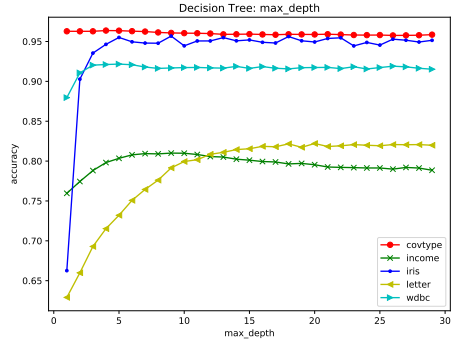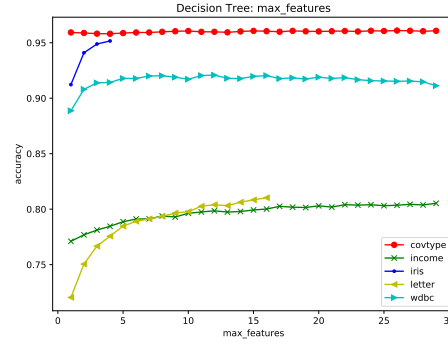
(a) KNN: k

(b) KNN: k

(c) Decision Tree: max_depth

(d) Decision Tree: $max\_features$

Figure 1: Validation score by hyperparameter value for KNN (1a and 1b) and Decision Tree (1c and 1d), averaged over data, shuffles, and train split. 1a zooms in on a subset of the results shown in 1c for greater detail. 1c depicts results for varying levels of max_depth, averaged over $max\_features$, wheres the opposite is done in 1c.

Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.
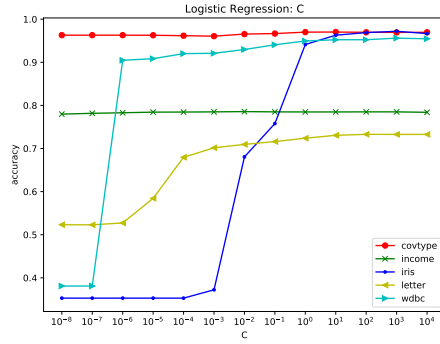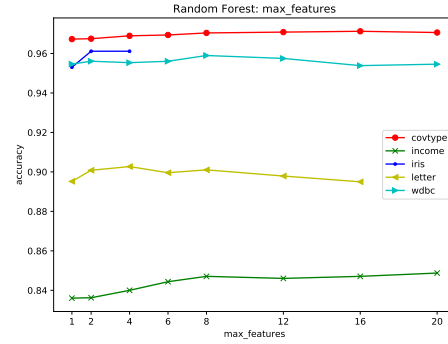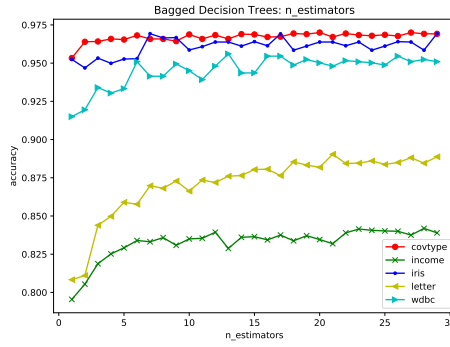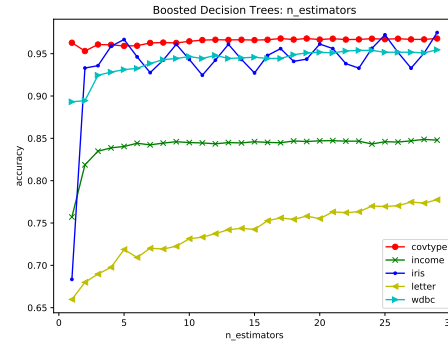
(a) Logistic Regression: C

(b) Random Forest: $n\_estimators$

(c) Bagged Decision Trees: $n\_estimators$

(d) Boosted Decision Trees: $n\_estimators$

Figure 2: Validation score by hyperparameter value for Logistic Regression (2a), Random Forest (2b), Bagged Decision Trees (2c), and Boosted Decision Trees (2d).

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.
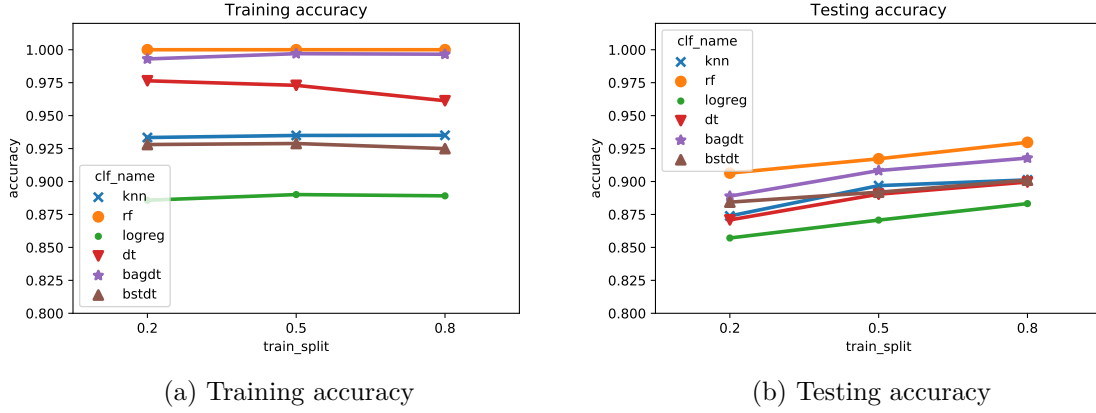
(a) Training accuracy

(b) Testing accuracy

Figure 3: Training (3a) and testing (3b) accuracy by train split and classifier, averaged over problems and shuffles (rf: Random Forest, logreg: Logistic Regression, dt: Decision Tree, bagdt: Bagged Trees, bstdt: Boosted Trees)

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Table 1: Best hyperparameter values after grid search by classifier and problem in each of three random shuffles (0.8 train split):

Classifier	Hyperparam.	Problem				
		COVTYPE	INCOME	IRIS	LETTER	WDBC
bagdt	$n\_estimators$	3,27,14	26,25,28	9,29,7	21,14,18	13,9,16
bstdt	$n\_estimators$	24,19,12	28,18,18	5,29,24	29,28,24	21,29,10
dt	max_depth	3,5,4	8,4,6	9,9,3	18,19,18	6,5,6
	$max\_features$	18,24,19	18,24,26	3,4,4	10,15,6	7,11,15
knn	$n\_neighbors$	4,5,9	8,6,17	12,9,1	1,1,1	9,7,7
logreg	C	10^2,1,10	1e-05,01,01	$10,10^3,10^2$	$10^2,10^3,10^2$	$10^3$0,10,10^3
rf	$max\_features$	20,16,16	20,16,8	1,2,2	4,2,8	12,8,8
	$n\_estimators$	$2^{10},2^{10},2^{10}$	$2^{10},2^{10},2^{10}$	$2^{10},2^{10},2^{10}$	$2^{10},2^{10},2^{10}$	$2^{10},2^{10},2^{10}$

4

Table 2: Classifier testing accuracy by problem, averaged over shuffles (0.8 train split):

Classifier	WDBC	INCOME	IRIS	COVTYPE	LETTER
Bagged Trees	.968	.833	.944	.971	.873
Boosted Trees	.968	.840	.956	.971	.771
Decision Tree	.953	.799	.956	.970	.820
KNN	.950	.757	.933	.967	**.899**
Logistic Regression	.968	.792	**.967**	.968	.721
Random Forest	**.982**	**.848**	.944	**.976**	.897

Table 3: Ranked classifiers by testing accuracy, averaged over problems and shuffles (0.8 train split):

Classifier	Accuracy
Random Forest	.930
Bagged Trees	.918
KNN	.901
Boosted Trees	.901
Decision Tree	.900
Logistic Regression	.883

4. Conclusion

5. Bonus Points

6. References

Appendix A.