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

2.1 Problems

Table 1: Original and limited sizes of data sets

Data set	Orig. Size	Lim. Size
COVTYPE	(581011×55)	(2000×55)
INCOME	(32561×109)	(2000×109)
LETTER	(20000×16)	(2000×16)
IRIS	— (150×5) —	
WDBC	— (569×31) —	

3. Experiments

3.1 Hyperparameters and Performance

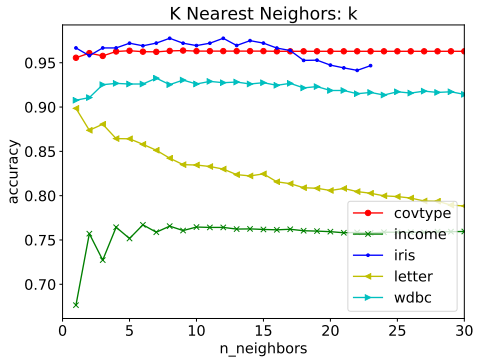
3.2 Train/Test split and Performance

3.3 Overall Classifier Performance

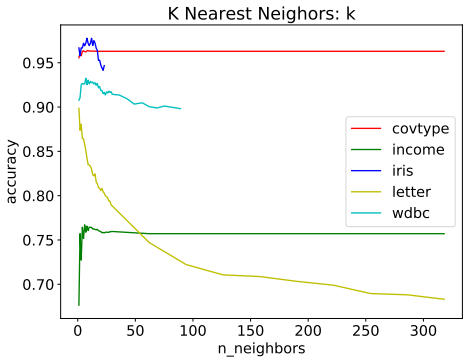
4. Conclusion

5. Bonus Points

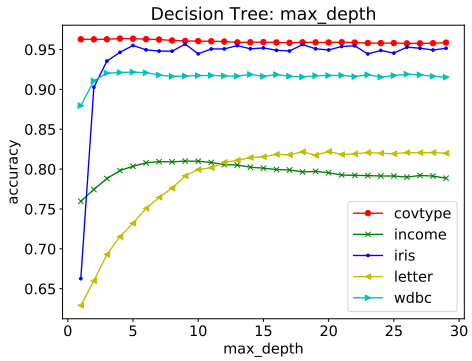
6. References



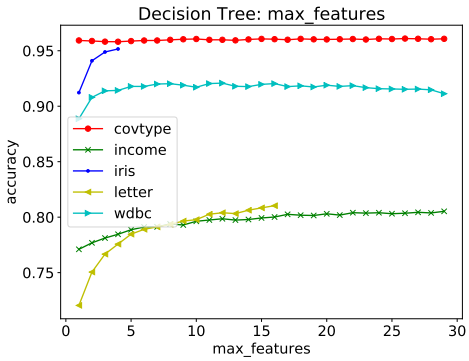
(a) KNN: k



(b) KNN: k



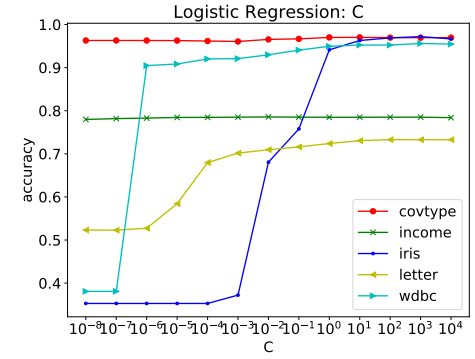
(c) Decision Tree: *max_depth*



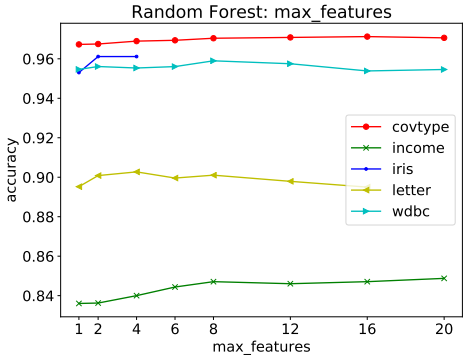
(d) Decision Tree: *max_features*

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.

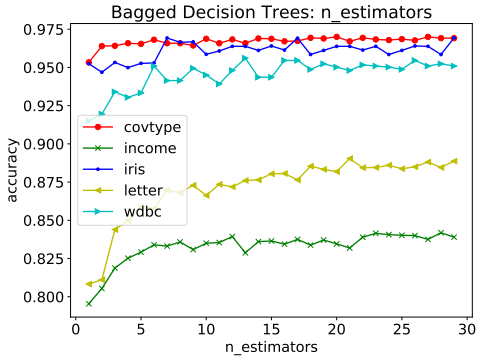
Appendix A.



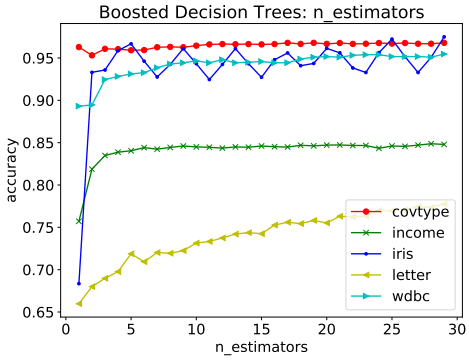
(a) Logistic Regression: C



(b) Random Forest: *n_estimators*

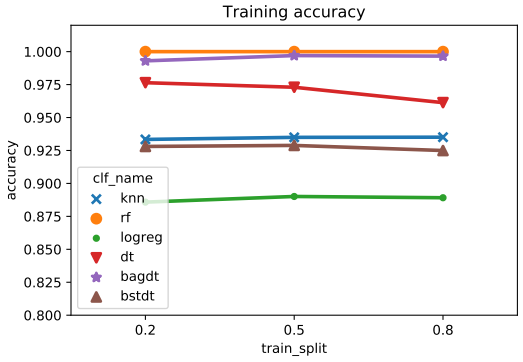


(c) Bagged Decision Trees: *n_estimators*

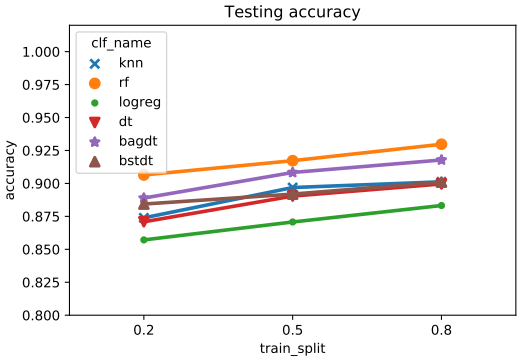


(d) Boosted Decision Trees: *n_estimators*

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



(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)

Table 2: Best hyperparameter values after grid search by classifier and problem in each of three random shuffles (0.8 train split):

		Problem				
Classifier	Hyperparam.	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}

Table 3: 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 4: 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