Weighted Random Forests

MID 2022

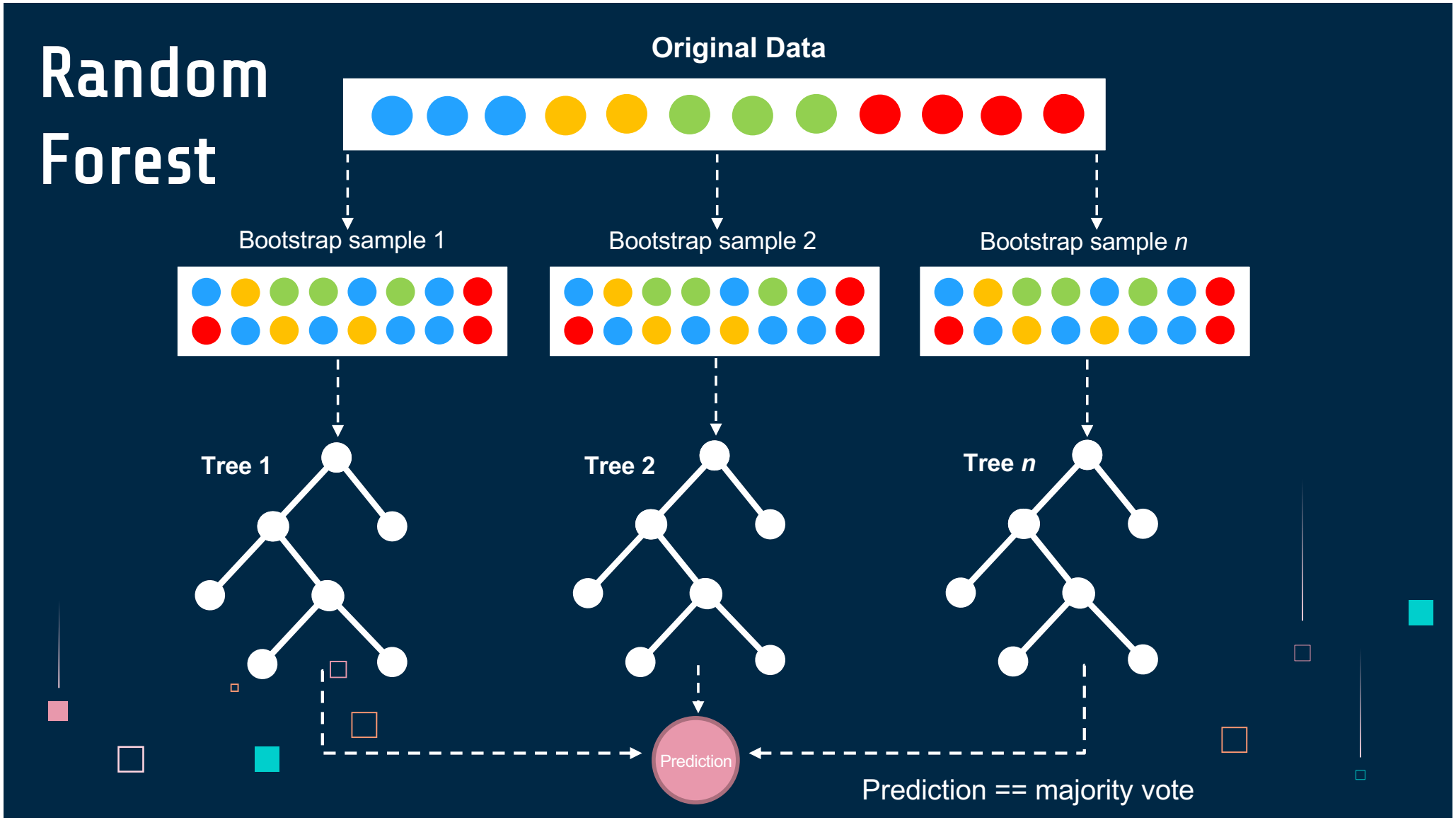
December 9, 2022

Ray Jennings

rj07609p@pace.edu

Traditional Random Forest Classifier

- *n* Weak Learner Trees
- Each tree uses sample data with replacement (bootstrap)
- Fit each tree with its “bootstrap” sample data
- Use majority voting across the *n* trees for final predicted value

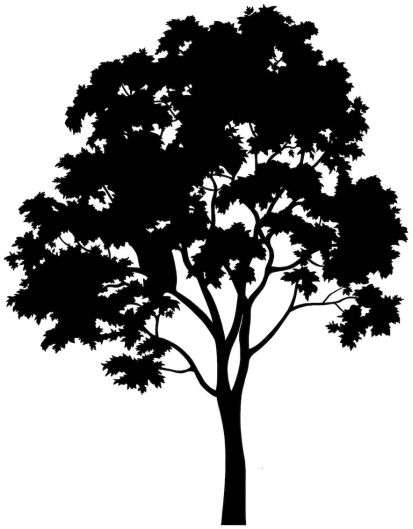

Random Forest
Original Data
Bootstrap sample 1
Bootstrap sample 2
Bootstrap sample n
Tree 1
Tree 2
Tree n
Prediction
Prediction == majority vote

Not all trees are created equal…

During the *fit* (training) phase, each weak tree will be created differently based on the sampled data….

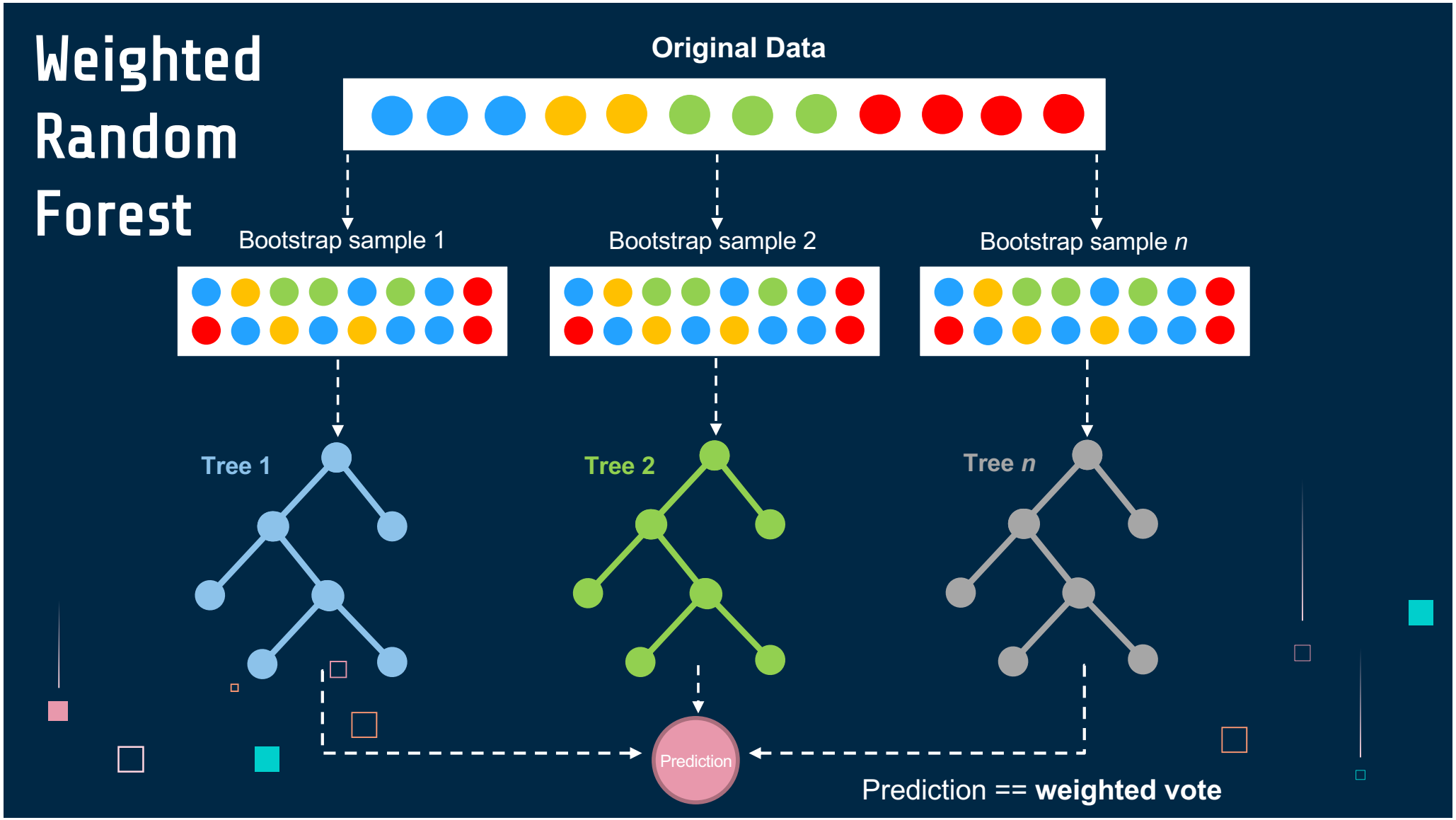
Smart Tree

Mediocre Tree



Dumb Tree

Let the smarter trees have more weight in their vote.


Weighted Random Forest
Original Data
Bootstrap sample 1
Bootstrap sample 2
Bootstrap sample n
Tree 1
Tree 2
Tree n
Prediction
Prediction == weighted vote

Prior Work

[1] Robnik-Šikonja, M. (2004), "Improving Random Forests," Machine Learning: ECML 2004. ECML 2004. Lecture Notes in Computer Science(), vol 3201. Springer, Berlin, Heidelberg.

Uses only a subset of the weak learners where the subset is chosen based on a similarity measure between instances already seen.

[2] M. El Habib Daho, N. Settouti, M. El Amine Lazouni and M. El Amine Chikh, "Weighted vote for trees aggregation in Random Forest," *2014 International Conference on Multimedia Computing and Systems (ICMCS)*, 2014, pp. 438-443, doi: 10.1109/ICMCS.2014.6911187.

Weight each of the weak learners by the performance based on the OOB

[3] S. Cha, Pace University, CS 655 lecture notes.

Use weighted voting by calculating the importance of each weak learner and use the value as the weight for each weak learner:

$$\alpha_i = \frac{1}{2}(\ln a_i - \ln \varepsilon_i)$$

Determine How Smart every Tree is and Assign Weights

- Determine the accuracy of each weak learner by its accuracy during the validation phase
- Weight each weak learner by its accuracy, for example:
- $weight = accuracy^p$ (or other formula)
- As the value of p increases the higher accuracies are favored and the lower accuracies favored less. Typical range: $1 \leq p \leq 10$
- To determine the point where, for a given value of p, higher accuracies are favored: $(accuracy^p)$
- Find where: $d\frac{x^p}{dx} = 1$, $\sqrt[p-1]{\frac{1}{p}} = 1$

$p = 2$	0.5000
$p = 3$	0.5774
$p = 4$	0.6270
$p = 5$	0.6687
$p = 10$	0.7743

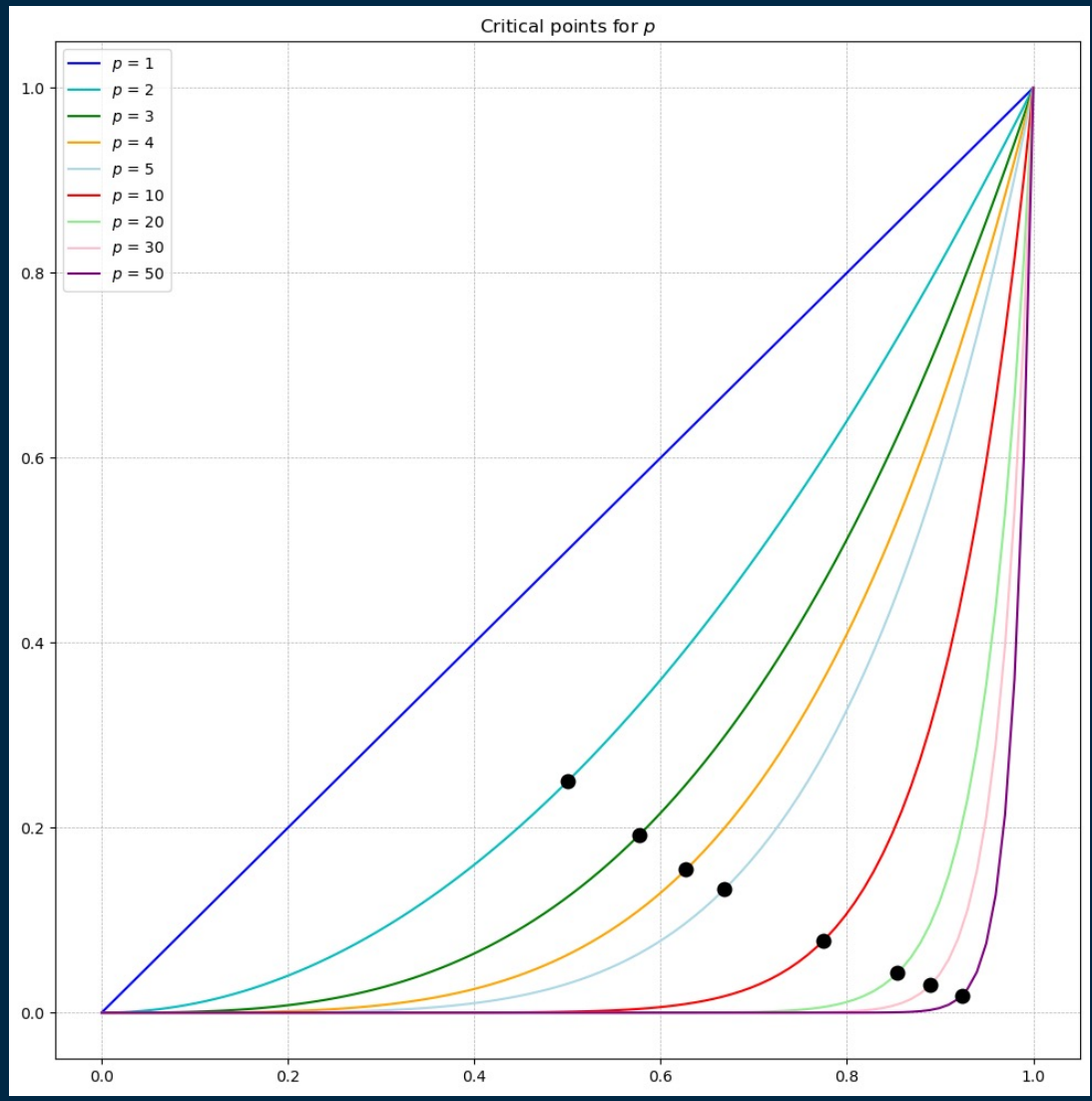
$p = 0$ resets to unweighted majority voting
$p = 1$ uses the accuracy as the weights
$p >> 10$ not particularly useful

$accuracy^p$

Find optimal p value based on accuracies.

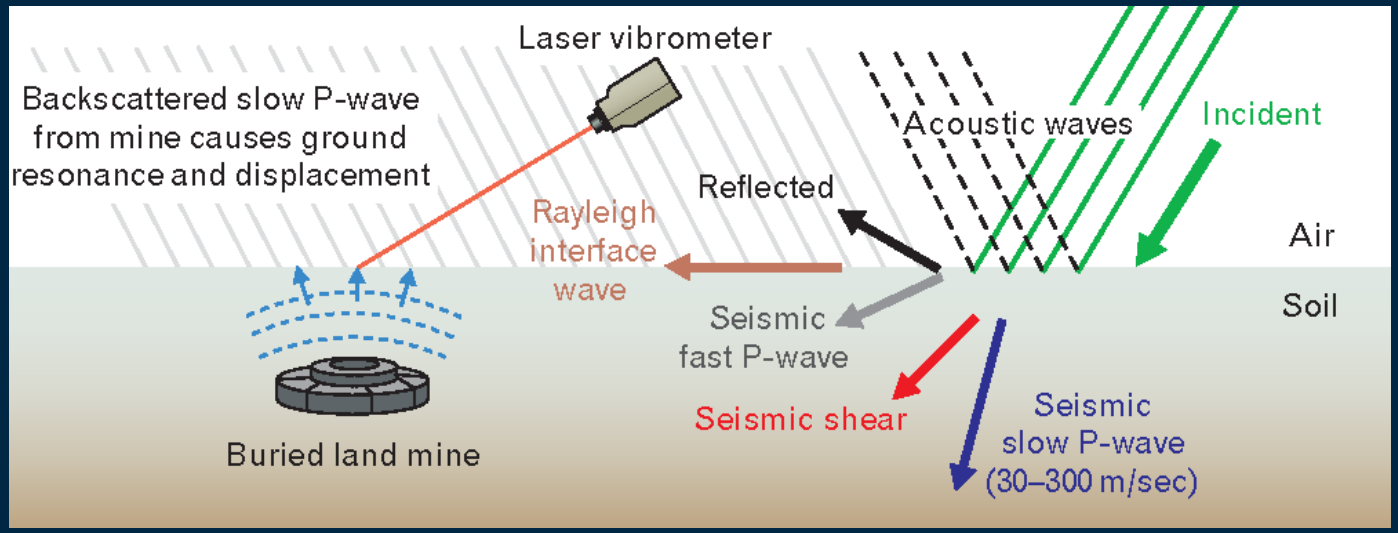
Higher values for p will favor higher accuracies.

p needs to be in the correct range.



SONAR – Rock or Mine

http://archive.ics.uci.edu/ml/datasets/connectionist+bench+(sonar,+mines+vs.+rocks)



Use Sonar readings to predict either a rock or mine

Dimensionality of 60: numbers in the range 0.0 to 1.0

Dataset size of 208 samples

Each number represents the energy within a particular frequency band, integrated over a certain period of time

Target feature is either: *Rock* or *Mine*

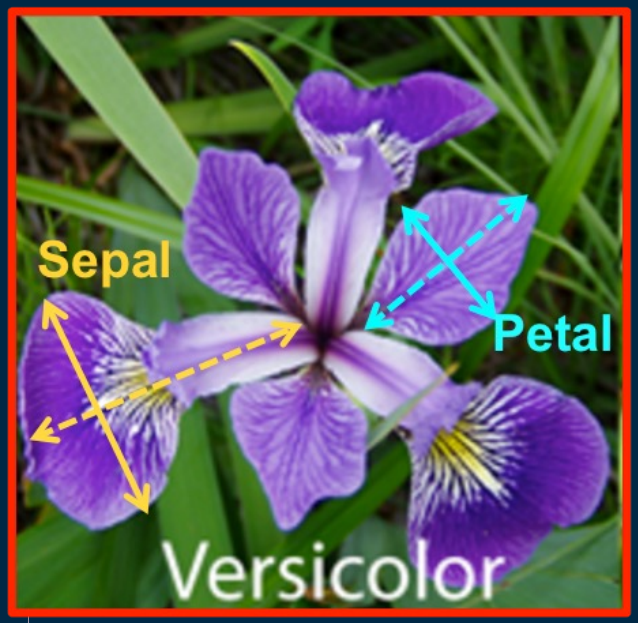
Palmer Penguin Species https://archive-beta.ics.uci.edu/dataset/690/palmer+penguins-3

- Data comes from 3 penguin species in the islands of Palmer Archipelago, Antarctica.

Species, Region, Island, Stage, Clutch Completion, Date Egg, Culmen Length (mm), Culmen Depth (mm), Flipper Length (mm), Body Mass (g), Sex

* Dropped a number of columns and use species as the target feature.

Iris Flowers https://archive.ics.uci.edu/ml/machine-learning-databases/iris/



Red Wine Quality https://archive.ics.uci.edu/ml/datasets/wine+quality

Dataset:

Red wine variants of the Portuguese "Vinho Verde" wine.

* Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available. There is no data about grape types, wine brand, wine selling price.

fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, quality (target feature)

White Wine Quality https://archive.ics.uci.edu/ml/datasets/wine+quality

Dataset:

White wine variants of the Portuguese "Vinho Verde" wine.

* Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available. There is no data about grape types, wine brand, wine selling price.

fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, quality (target feature)

RESULTS

Sonar - Rock or Mine *(p = 5)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			*10 Trees*			*15 Trees*			*20 Trees*			*25 Trees*		
	RF	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***
Fold 1	0.82927	0.87805	0.78049	0.82927	0.80488	0.85366	0.68293	0.78049	0.87805	0.82927	0.85366	0.85366	0.82927	0.80488	0.78049
Fold 2	0.78049	0.78049	0.73171	0.85366	0.87805	0.87805	0.78049	0.82927	0.78049	0.73171	0.7561	0.85366	0.82927	0.80488	0.87805
Fold 3	0.65854	0.78049	0.70732	0.78049	0.82927	0.65854	0.80488	0.85366	0.85366	0.80488	0.85366	0.78049	0.82927	0.92683	0.7561
Fold 4	0.80488	0.80488	0.73171	0.78049	0.90244	0.80488	0.73171	0.85366	0.7561	0.78049	0.82927	0.80488	0.7561	0.78049	0.7561
Fold 5	0.78049	0.82927	0.87805	0.78049	0.78049	0.73171	0.82927	0.80488	0.78049	0.85366	0.97561	0.82927	0.82927	0.87805	0.82927
Avg	0.77073	**0.81464**	***0.76586***	0.80488	**0.83903**	***0.78537***	0.76586	**0.82439**	***0.80976***	0.80000	**0.85366**	***0.82439***	0.81464	**0.83903**	***0.80000***
Winner		***X***			***X***			***X***			***X***			***X***	

RESULTS

Sonar - Rock or Mine *(p = 10)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			*10 Trees*			*15 Trees*			*20 Trees*			*25 Trees*		
	RF	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***
Fold 1	0.80488	0.82927	0.82927	0.78049	0.82927	0.80488	0.75610	0.78049	0.80488	0.80488	0.82927	0.75610	0.78049	0.90244	0.92683
Fold 2	0.85366	0.85366	0.73171	0.80488	0.82927	0.75610	0.75610	0.87805	0.82927	0.90244	0.92683	0.82927	0.75610	0.85366	0.82927
Fold 3	0.78049	0.80488	0.85366	0.73171	0.80488	0.80488	0.90244	0.90244	0.90244	0.80488	0.87805	0.85366	0.85366	0.85366	0.90244
Fold 4	0.75610	0.78049	0.68293	0.75610	0.80488	0.78049	0.73171	0.80488	0.82927	0.82927	0.82927	0.80488	0.73171	0.82927	0.78049
Fold 5	0.80488	0.82927	0.82927	0.75600	0.85366	0.85366	0.78049	0.85366	0.85366	0.78049	0.90244	0.90244	0.68293	0.95122	0.78049
Avg	0.80000	**0.81951**	***0.78537***	0.76584	**0.82927**	***0.80000***	0.78537	**0.84390**	***0.84390***	0.82439	**0.87317**	***0.82927***	0.76098	**0.87805**	***0.84390***
Winner		***X***			***X***			***X***			***X***			***X***	

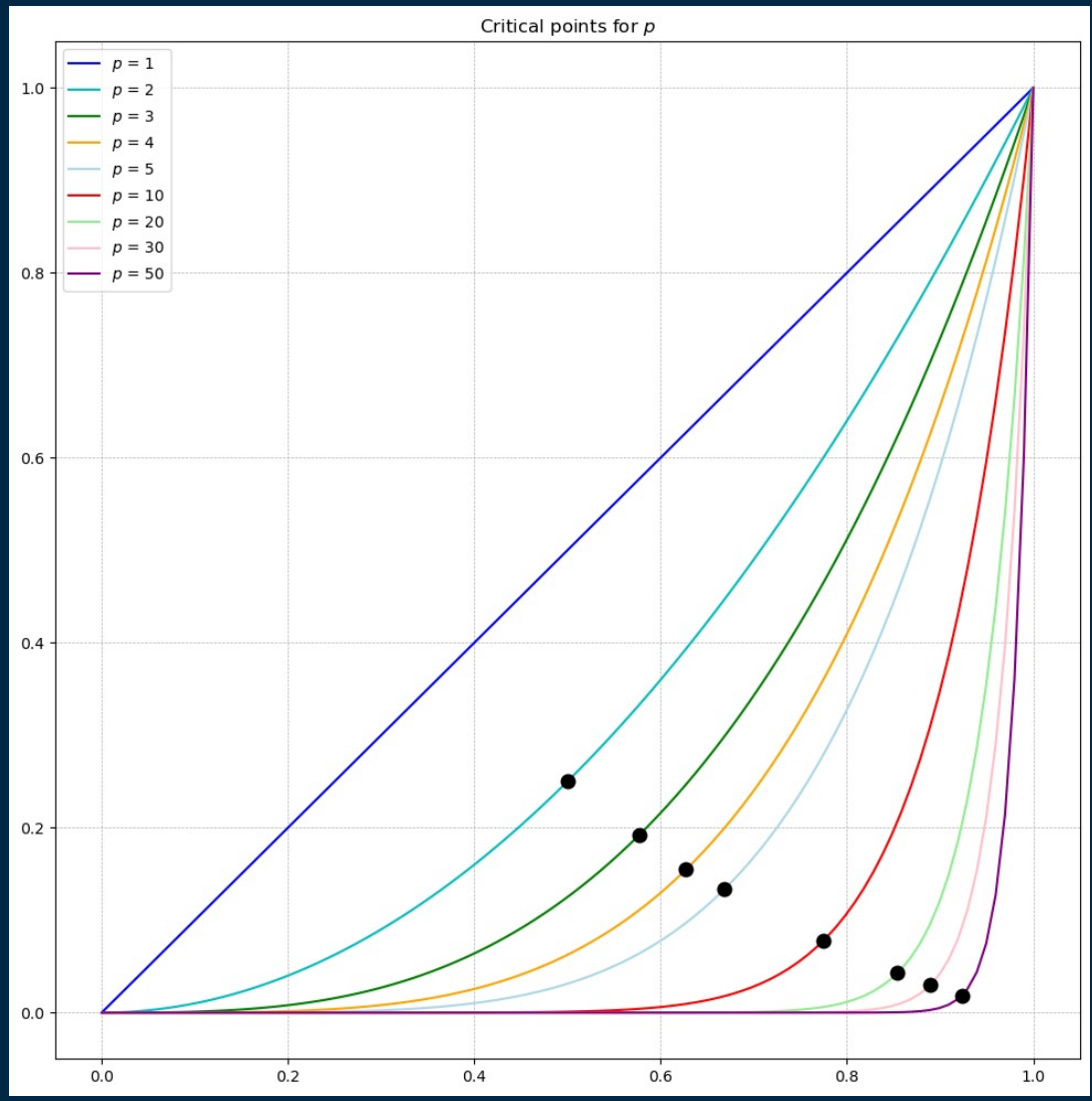
- Improvement over *p = 5 : 0.005, -0.010, 0.012, 0.012, 0.039*

$accuracy^p$

Find optimal p value based on accuracies.

Higher values for p will favor higher accuracies.

p needs to be in the correct range.



Extreme Case

Palmer Penguins *(p = 50)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			*10 Trees*			*15 Trees*			*20 Trees*			*25 Trees*		
	RF	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***
Fold 1	0.96970	0.98485	0.98485	0.96970	0.98485	0.98485	0.98485	0.98485	1.00000	0.96970	0.98485	0.98485	1.00000	1.00000	0.96970
Fold 2	1.00000	0.98485	1.00000	0.95455	0.98485	0.98485	0.93939	0.95455	0.98485	1.00000	0.98485	0.98485	0.95455	0.96970	0.98485
Fold 3	0.98485	0.98485	1.00000	0.98485	0.98485	1.00000	0.96970	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
Fold 4	0.95455	0.96970	1.00000	0.95455	0.98485	1.00000	0.98485	1.00000	1.00000	0.98485	0.98485	0.98485	1.00000	1.00000	1.00000
Fold 5	0.95455	0.95455	1.00000	0.98485	0.98485	1.00000	0.98485	1.00000	0.98485	1.00000	1.00000	1.00000	0.98485	1.00000	1.00000
Avg	0.97273	**0.97576**	0.99697	0.96970	**0.98485**	0.99394	0.97273	**0.98788**	0.99394	0.99091	0.99091	0.99091	0.98788	**0.99394**	***0.99091***
Winner		***X***			***X***			***X***						***X***	

- Base accuracies are already > 96%, so need very high value of p to improve: $p \geq 50$
  - Improvement over $p = 10$: *-0.0059, 0.0064, 0.0033, 0.0063, 0.0093*

Extreme Case

Palmer Penguins *(p = 100)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			*10 Trees*			*15 Trees*			*20 Trees*			*25 Trees*		
	RF	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***
Fold 1	0.95455	0.95455	0.98485	0.98485	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.96970	0.98485	0.98485
Fold 2	0.96970	0.96970	0.98485	1.00000	0.96970	1.00000	0.95455	0.98485	1.00000	0.98485	0.98485	0.96970	1.00000	1.00000	1.00000
Fold 3	0.96970	0.96970	1.00000	0.95455	0.96970	1.00000	0.98485	0.98485	0.98485	0.98485	0.98485	0.98485	1.00000	1.00000	0.96970
Fold 4	0.95455	0.95455	0.95455	0.96970	0.96970	0.96970	1.00000	0.98485	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
Fold 5	1.00000	1.00000	1.00000	0.96970	0.95455	0.98485	1.00000	1.00000	1.00000	0.98485	0.98485	1.00000	0.98485	0.98485	0.98485
Avg	0.96970	0.96970	0.98485	0.97576	0.97273	0.99091	0.98788	**0.99091**	0.99697	0.99091	0.99091	0.99091	0.99091	**0.99394**	***0.98788***
Winner								***X***						***X***	

Improvement over *p = 10 : -0.0118, -0.0057, 0.0063, 0.0063, 0.0093*

Results

Red Wine Quality *(p = 2)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			10 Trees			15 Trees			20 Trees			25 Trees		
	RF	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***	***RF***	***WRF***	***SKL***
Fold 1	0.57053	0.57994	0.58307	0.66458	0.67398	0.66771	0.63636	0.65204	0.64263	0.68652	0.68966	0.67398	0.69279	0.68966	0.68652
Fold 2	0.62069	0.62069	0.65831	0.63950	0.65517	0.70219	0.67712	0.68025	0.68652	0.64263	0.64263	0.66144	0.67398	0.67085	0.67712
Fold 3	0.62069	0.63950	0.67085	0.64263	0.63323	0.64890	0.65204	0.65517	0.67712	0.67398	0.67398	0.64577	0.67398	0.68025	0.65831
Fold 4	0.61129	0.61442	0.63009	0.65517	0.68339	0.67398	0.68652	0.68339	0.68966	0.63636	0.63009	0.67085	0.64890	0.65831	0.67398
Fold 5	0.63009	0.62069	0.62069	0.65204	0.64890	0.69906	0.67085	0.66771	0.67085	0.65831	0.65517	0.68652	0.66458	0.67085	0.66144
Avg	0.61066	**0.61505**	0.63260	0.65078	**0.65893**	0.67837	0.66458	**0.66771**	0.67335	0.65956	0.65831	0.66771	0.67085	**0.67398**	***0.67147***
Winner		***X***			***X***			***X***						***X***	

Most Complex

White Wine Quality *(p = 3)*

RF: *Base Random Forest*
WRF: *Weighted Random Forest*
SKL: *Scikit-Learn Random Forest Classifier*

	5 Trees			*10 Trees*			*15 Trees*			*20 Trees*			*25 Trees*		
	RF	*WRF*	*SKL*	*RF*	*WRF*	*SKL*	*RF*	*WRF*	*SKL*	*RF*	*WRF*	*SKL*	*RF*	*WRF*	*SKL*
Fold 1	0.58018	0.58325	0.60674	0.60981	0.60776	0.63432	0.61798	0.62104	0.66905	0.58938	0.59040	0.63534	0.60266	0.60368	0.62717
Fold 2	0.57712	0.57916	0.60776	0.60674	0.62002	0.65986	0.60163	0.60061	0.63841	0.59448	0.59551	0.61389	0.63534	0.63534	0.64045
Fold 3	0.56282	0.56180	0.60470	0.61696	0.61389	0.60981	0.58325	0.58223	0.62819	0.60674	0.61287	0.63739	0.60776	0.60572	0.63228
Fold 4	0.59040	0.59040	0.60368	0.59040	0.59551	0.59653	0.60878	0.61083	0.62308	0.60878	0.60981	0.62819	0.59653	0.59653	0.62308
Fold 5	0.59040	0.58938	0.65066	0.59448	0.60163	0.60368	0.62104	0.62308	0.63943	0.61083	0.60776	0.64760	0.62104	0.62615	0.64249
Avg	0.58018	**0.58080**	0.61471	0.60368	**0.60776**	0.62084	0.60654	**0.60756**	0.63963	0.60204	**0.60327**	0.63248	0.61267	**0.61348**	0.63309
Winner		***X***			***X***			***X***			***X***			***X***	

Future Work

- Dynamically estimate the best weight calculation (value for ρ or other weight formula)
- Improve the execution runtime of Weighted Random Forest
- Save a model such that it can be used again with the weights saved as part of the model

Thank You

Ray Jennings
rj07609p@pace.edu