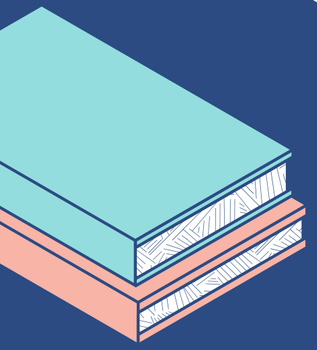
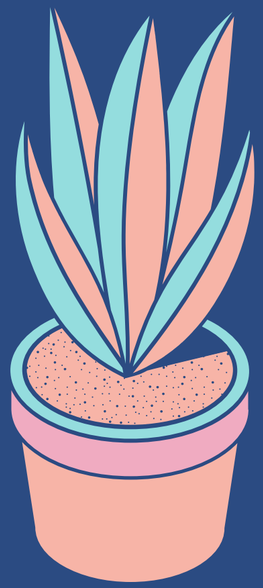


REPORT WEEK 3

Bagging for Horse Racing in HK

Le Viet Hung



Content

1 Dataset

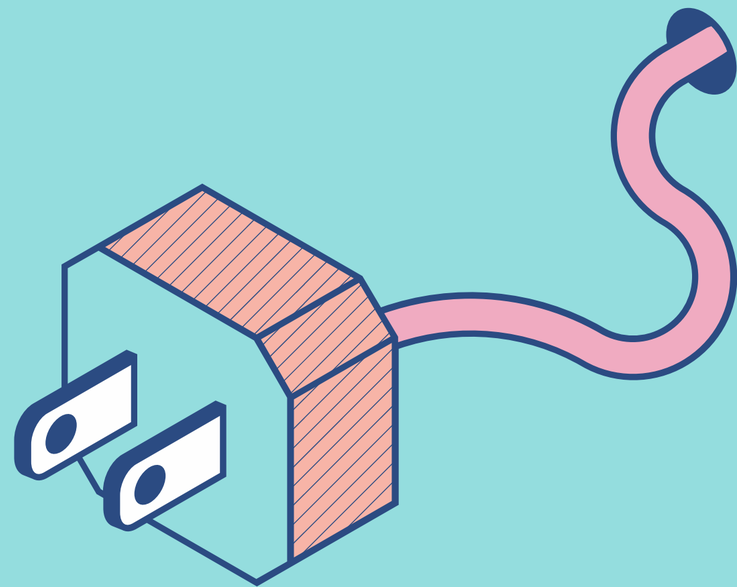
2 Feature Engineering

3 Model

4 Experiment

1.Dataset

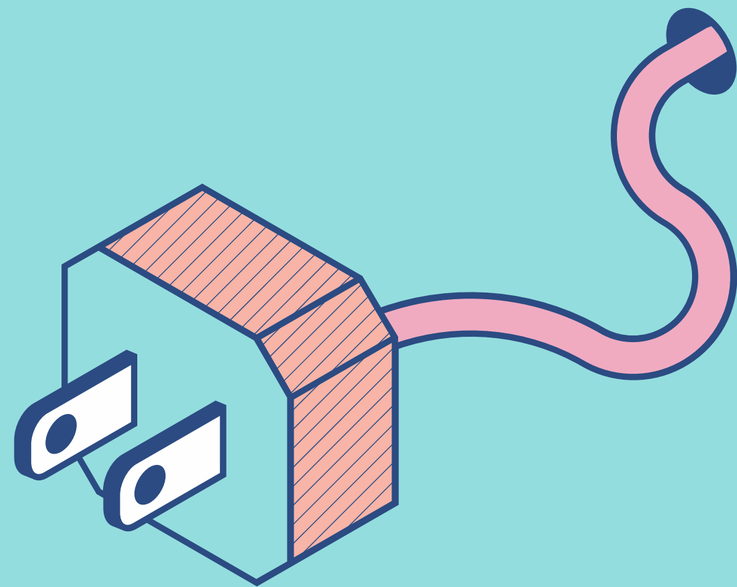
Hong Kong's horse racing industry is emphasized as a significant business with two race tracks in a relatively small area, boasting enormous betting pools, surpassing those of all US racetracks combined. The author is inviting data scientists to engage in this venture to tap into the potentially limitless opportunities for success in this unique market.



1.Dataset

This dataset included 2 part: race.csv , run.csv

- race.csv: Each line describes the condition of an individual race.
- run.csv: Each line describes the characteristics of one horse run, in one of the races given in races.csv.

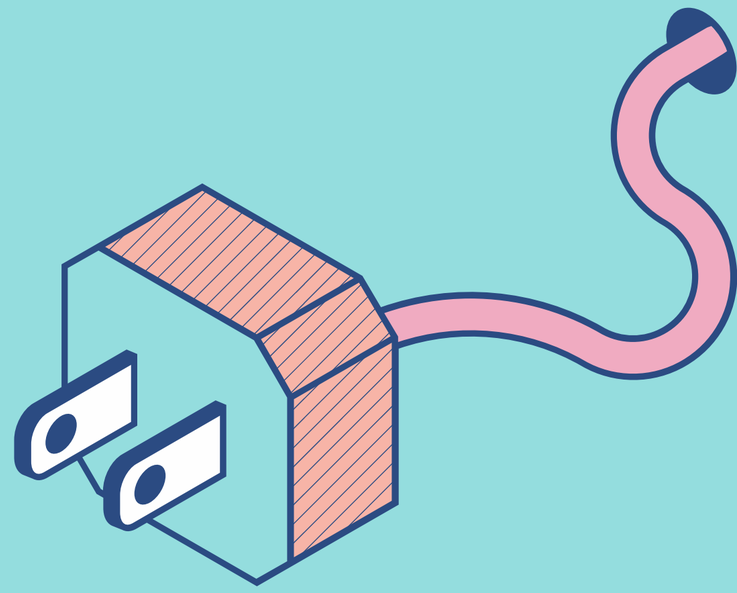


2.Feature Engineering

This dataset have a lot of features , but I just select some of the best feature.

race.csv: 'venue', 'config', 'surface', 'distance', 'going', 'race_class'

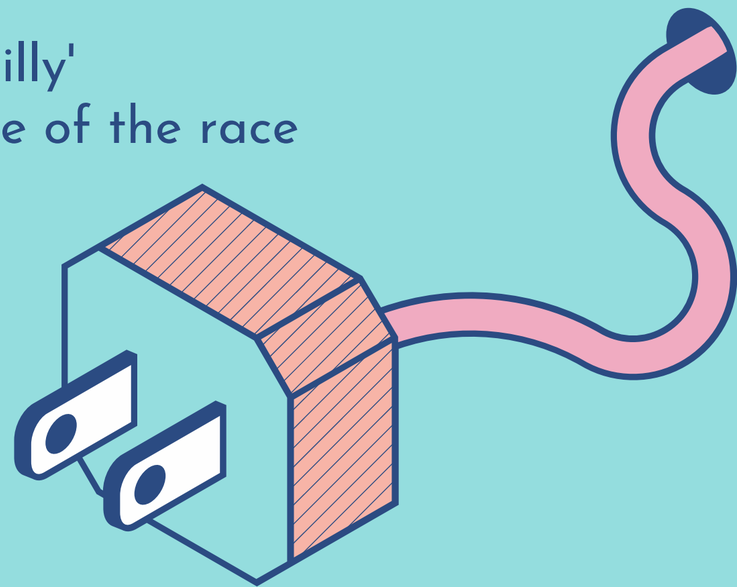
- venue: a 2-character string, representing which of the 2 race courses this race took place at: ST = Shatin, HV = Happy Valley
- config: race track configuration, mostly related to the position of the inside rail
- surface: a number representing the type of race track surface: 1 = dirt, 0 = turf
- distance: distance of the race, in metres
- going: track condition
- race_class: a number representing the class of the race



2.Feature Engineering

run.csv: 'race_id', 'draw', 'horse_age', 'horse_country', 'horse_type', 'horse_rating', 'declared_weight',actual_weight', 'win_odds', 'result'

- race_id: unique identifier for the race
- draw: post position number of the horse in this race
- horse_age: current age of this horse at the time of the race
- horse_country: country of origin of this horse
- horse_type: sex of the horse, e.g. 'Gelding', 'Mare', 'Horse', 'Rig', 'Colt', 'Filly'
- horse_rating: rating number assigned by HKJC to this horse at the time of the race
- declared_weight: declared weight of the horse and jockey, in lbs
- actual_weight: actual weight carried by the horse, in lbs
- win_odds: win odds for this horse at start of race
- results: finishing position of this horse in the race



2.Feature Engineering

Using Pivot to aggregate horses data from multiple rows, which belongs to a single race, into one row.

	race_id	draw	horse_age	horse_country	horse_type	horse_rating	declared_weight	actual_weight	win_odds	result
0	0	7	3	1	3	60	1020.0	133	9.7	10
1	0	12	3	11	3	60	980.0	133	16.0	8
2	0	8	3	11	3	60	1082.0	132	3.5	7
3	0	13	3	12	3	60	1118.0	127	39.0	9
4	0	14	3	5	3	60	972.0	131	50.0	6

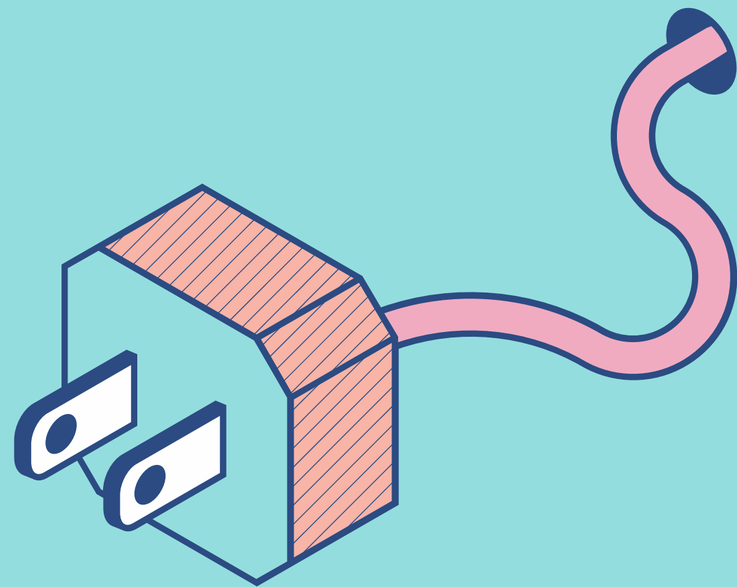
	horse_age	horse_country	horse_type	horse_rating	declared_weight	actual_weight	win_odds	horse_age	horse_country	horse_type	...										result
draw	1	1	1	1	1	1	1	2	2	2	...	5	6	7	8	9	10	11	12	13	14
race_id																					
0	3.0	14.0	3.0	60.0	1089.0	120.0	5.4	3.0	1.0	3.0	...	3.0	13.0	10.0	7.0	14.0	4.0	12.0	8.0	9.0	6.0
1	3.0	1.0	3.0	60.0	1059.0	121.0	10.0	3.0	11.0	3.0	...	8.0	2.0	5.0	10.0	12.0	1.0	3.0	9.0	14.0	13.0
2	3.0	1.0	3.0	60.0	1028.0	116.0	45.0	3.0	11.0	3.0	...	14.0	6.0	4.0	7.0	9.0	10.0	5.0	11.0	3.0	12.0
3	3.0	14.0	5.0	60.0	1074.0	115.0	2.9	3.0	11.0	3.0	...	3.0	11.0	8.0	9.0	6.0	12.0	2.0	10.0	0.0	0.0
4	3.0	11.0	3.0	60.0	988.0	106.0	31.0	3.0	11.0	3.0	...	14.0	6.0	2.0	8.0	10.0	13.0	7.0	4.0	11.0	12.0

2.Feature Engineering

Then, I use the **Encoder** to transform **a vector output to a numberical output**.

Top-1: Transform to the index of top-1
-> Example: [2 3 **1** 4 5 7 6 8 9 14 13 12 11 10] -> [0 0 **1** 0 0 0 0 0 0 0 0 0 0 0] transform to 3

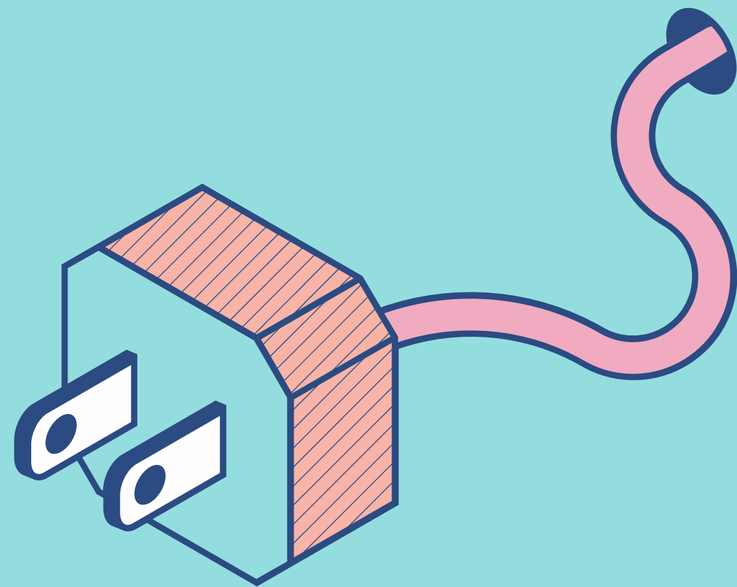
Top-3: Transform from binary array top-3 to decimal
-> Example: [**2 3 1** 4 5 7 6 8 9 14 13 12 11 10] -> [**1 1 1** 0 0 0 0 0 0 0 0 0 0 0] transform to 7



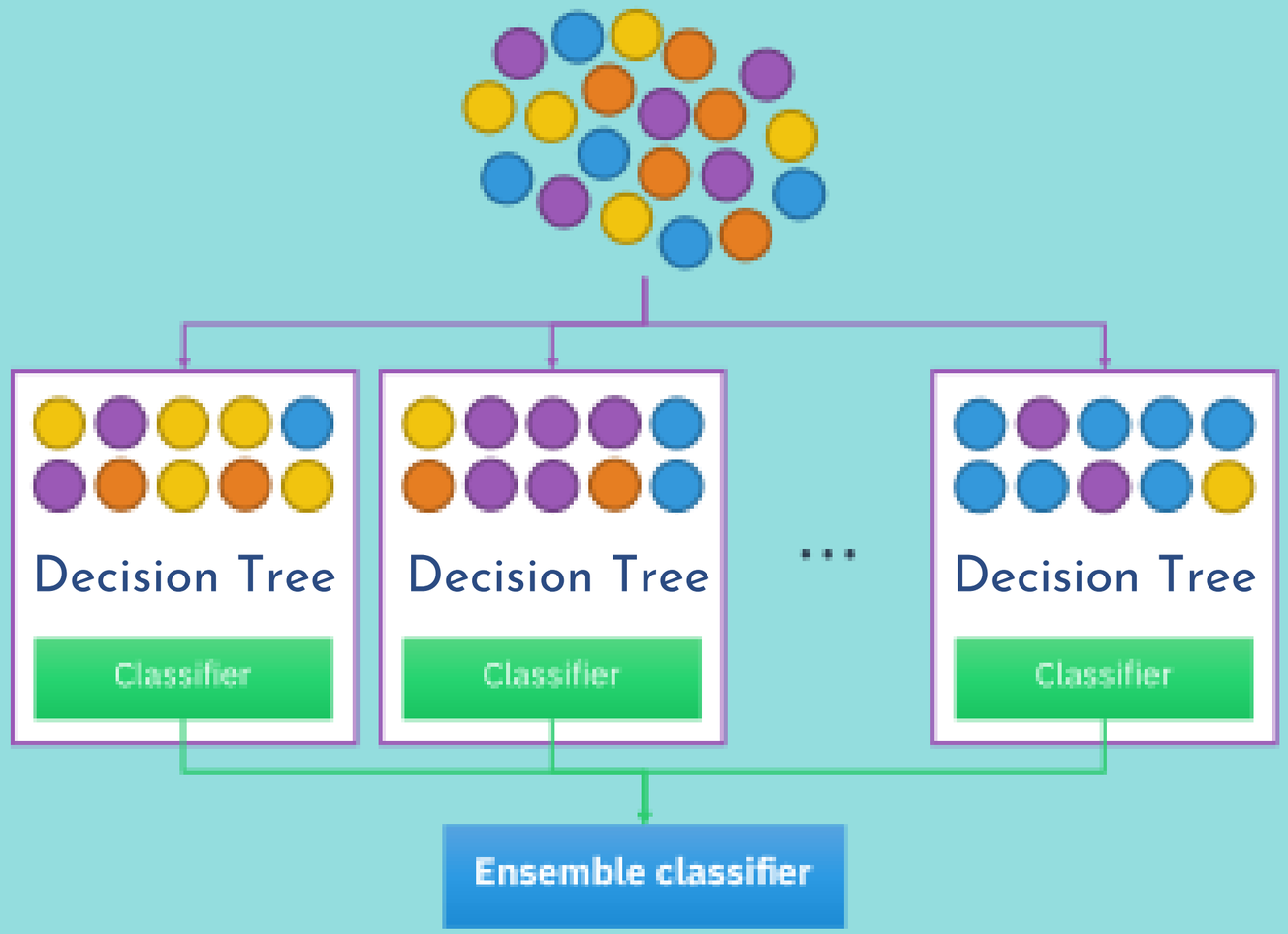
3.Model

For this problem, I use Ensemble Machine Learning method
That's bagging method, that often considers weak learners, learns them independently from each other in parallel and combines them following some kind of deterministic averaging process

In this prediction model, I used **Bagging algorithm for multiple Decision Tree**



3.Model

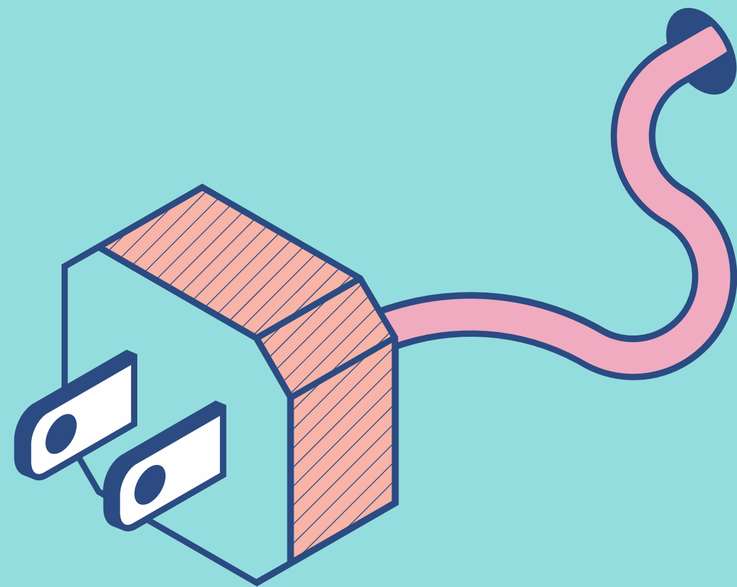


Original Data

Bootstrapping

Aggregating

Bagging

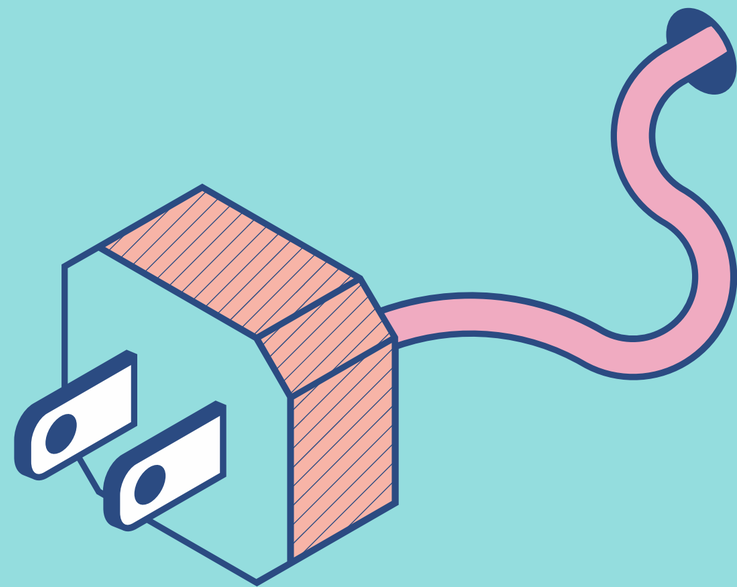


3.Model

For experiment, I use **Bagging with 50 Decision Tree**, bootstrap is True

Setting:

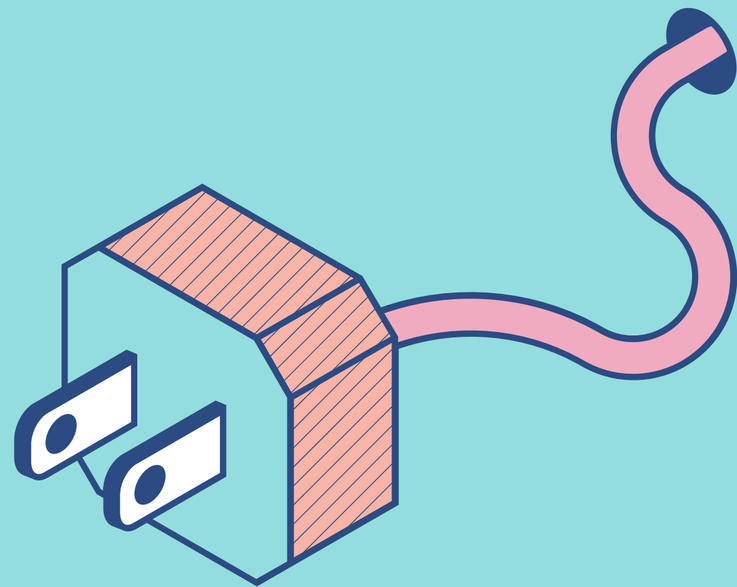
```
clf = DecisionTreeClassifier()
bag_clf = BaggingClassifier(clf, n_estimators=50,bootstrap=True,n_jobs=-1)
```



4.Experiment

Results: In this experiment, I use the metric of accuracy

	Accuracy
Prediction Top-1	**42.56%**
Prediction Top-3	**1,1%**



Thank you