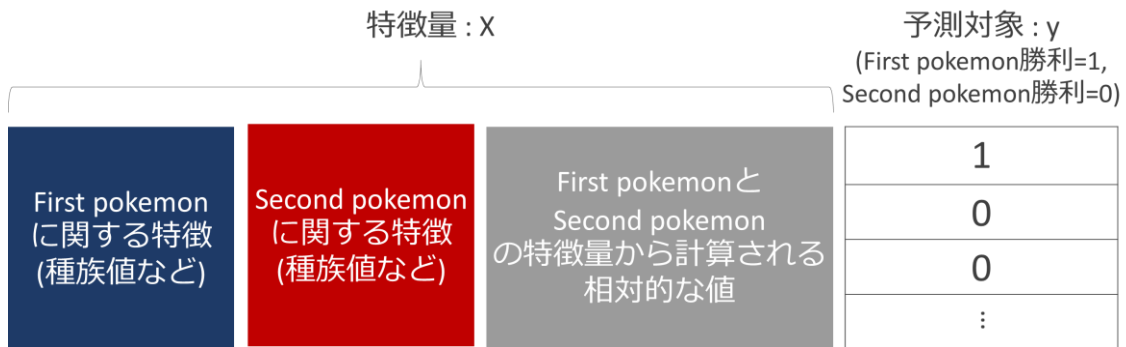


第 2 回 GCI コンペ

化学システム工学専攻修士 2 年 菅原悠樹

手法概要 / 解析の方針

ポケモンバトルで人が勝敗を予測する場合に考慮しそうな要素を特徴量に落とし込み、LightGBM(GBDT の一種)で予測。



作った matrix の概要 (図 1)

作成特徴量

1. 種族値そのまま
2. Type や世代、generation に関して one-hot-encoding
3. Pokemon の名前、Type や世代、generation に関して likelihood encoding
4. 対戦ポケモンでの Type 相性による First pokemon から Second pokemon への攻撃倍率, Second pokemon から First pokemon への攻撃倍率
(参考:<https://yakkun.com/data/aisyou.htm>)
5. 攻撃力と特殊攻撃の最大値、「攻撃力 - 相手の防御力」と「特殊攻撃力 - 相手の特殊防御」の大きい方
6. 以上の変数のグループについて、自分と相手の変数での交差項を作成

Tips

One-hot-encoding

k 種類の値を持つ categorical 変数に対して、それぞれの値に対応する k 列を用意して、自分の categorical の値のみを 1 にして他は 0 にする。(図 2)

Likelihood-encoding

Categorical 変数に対して、自分と同じ categorical 値を持つサンプルの予測対象 y の平均値に置き換える。(図 3)

ポケモン	One-hot-encoding	ゼニガメ	フシギダネ	ヒトカゲ
ゼニガメ		1	0	0
フシギダネ		0	1	0
ゼニガメ		1	0	0
ヒトカゲ		0	0	1
ヒトカゲ		0	0	1
ヒトカゲ		0	0	1
フシギダネ		0	1	0

図 2 One Hot Encoding

ポケモン	y	likelihood-encoding	P(1)
ゼニガメ	0		0
フシギダネ	0		0.5
ゼニガメ	0		0
ヒトカゲ	1		0.67
ヒトカゲ	1		0.67
ヒトカゲ	0		0.67
フシギダネ	1		0.5

図 3 Likelihood Encoding

感想

likelihood encoding で実装が少し間違っていたことに、これを書いていて気づきました。(First pokemon と Second pokemon で別々に encoding してしまっている。First と Second の結果を各ポケモン毎に統合して encoding するべき)

速さに関する特徴の重要度が高かった(気がした)。ポケモン詳しい人が、その知見に基づいて解析すればもっと精度上がりそう。

速さが重要なら、各 pokemon について先制攻撃確定技を持たされることが多いかどうかについて、(ポケモン得意な人が) 特徴を作ってあげたらしたらいいなと思いましたが、ポケモンあまりやらないしさすがに面倒くさすぎてやらなかったです。

かなり特徴を作りこんだのですが、GBDT ではスコアが上がりましたがただの決定木(Extra Trees や Random Forest)で全くスコアが上がらず、？って感じです。