

No-Free-Lunch Theorem

Do que se trata?



Classificação de algoritmos

- Tarefa: dado um conjunto de exemplos pré-classificados, induzir um modelo/classificador para novos casos.
- Aprendizado Supervisionado: classes são conhecidas para os exemplos usados para construir o modelo/classificador
- Um classificador pode ser um conjunto de regras lógicas, uma árvore de decisão, uma rede neural, etc.
- Aplicações típicas: aprovação de crédito, marketing direto, detecção de fraude,

Classificação de algoritmos

- Algoritmos simples geralmente funcionam muito bem na prática. Além disso:
 - Menor tempo de construção do modelo;
 - Combinação (*ensembles*) de algoritmos simples;
- O sucesso de cada algoritmo é dependente do domínio de aplicação (***No Free Lunch Theorems***)

No Free Lunch Theorem em ML

“ Para quaisquer dois algoritmos de aprendizado, há tantas situações (apropriadamente ponderadas) nas quais o algoritmo 1 é superior ao algoritmo 2 e vice-versa, de acordo com qualquer uma das medidas de ‘superioridade’ “

No Free Lunch Theorem em ML

- Qualquer vantagem apresentada por um algoritmo sobre uma classe específica de problemas é compensada quando de sua aplicação em outra classe, no caso em que outro algoritmo resultará mais adequado.
- Se todos os problemas forem igualmente possíveis, então os algoritmos tenderão a apresentar, na média, o mesmo comportamento preditivo.
- A definição do algoritmo mais satisfatório a cada aplicação deve ser feita de maneira pontual.

Por que desenvolver novos algoritmos

- O “desenvolvedor” (você) está preocupado em escolher o algoritmo mais apropriado para o **problema em questão**
- Isso requer algum conhecimento *a priori* – distribuição de dados, probabilidades anteriores, complexidade do problema, a física do fenômeno subjacente, etc.

Por que desenvolver novos algoritmos

- O teorema do No Free Lunch nos diz que – a menos que tenhamos algum conhecimento *a priori* – classificadores simples (ou complexos) não são necessariamente melhores que outros. No entanto, dadas algumas informações *a priori*, certos classificadores podem melhorar as características de certos tipos de problemas.
- O principal desafio do “desenvolvedor” é, então, identificar a correspondência correta entre o problema e o classificador! ... O que é mais um motivo para se armar com um conjunto diversificado de arsenal de aprendiz!