

Machine Learning Package

Portfolio de algoritmos de Machine Learning

Sumário

- Os conjuntos de modelos (*ensemble*) têm como base a combinação das previsões de vários modelos.
- Iremos implementar um modelo *ensemble* que usa a votação como função de combinação – *VotingClassifier*

Datasets

- Os datasets estão disponíveis em:
 - <https://www.dropbox.com/sh/oas4yru2r9n61hk/AADpRunbqES44W49gx9deRN5a?dl=0>

Objeto *VotingClassifier*

- Adiciona ao sub-package *ensemble* o modulo *voting_classifier.py* com o objeto *VotingClassifier*.
- *class VotingClassifier*:
 - Parâmetros:
 - *models* – conjunto de modelos
 - Métodos:
 - *fit* – treina os modelos do *ensemble*
 - *predict* – estima a variável de saída usando os modelos treinados e a função de combinação
 - *score* – calcula o erro entre as previsões e os valores reais

Objeto *VotingClassifier*

- *VotingClassifier.fit*:

1. Treina o conjunto de modelos
2. Retorna ele próprio

- *VotingClassifier.predict*:

1. Obtém previsões de cada modelo
2. Combina as previsões de cada modelo usando a técnica de votação. A classe mais representada deve ser a escolhida

Objeto *VotingClassifier*

- *VotingClassifier.score*:

1. Estima os valores de saída (Y) usando os modelos treinados
2. Calcula a *accuracy* entre os valores reais e as previsões

Teste *VotingClassifier*

- *VotingClassifier*:

1. Usa o dataset *breast-bin.csv*
2. Usa o *sklearn.preprocessing.StandardScaler* para standardizar os dataset.
`breast_dataset.X = StandardScaler().fit_transform(breast_dataset.X)`
3. Divide o dataset em treino e teste
4. Cria o modelo *KNNClassifier*
5. Cria o modelo *LogisticRegression*
6. Cria o modelo *ensemble VotingClassifier* usando os classificadores anteriores
7. Treina o modelo. Qual o score obtido?

Avaliação

- Exercício 6: Adiciona o modelo *ensemble StackingClassifier*
 - O modelo *StackingClassifier* usa um conjunto de modelos para gerar previsões. Estas previsões são depois usadas para treinar outro modelo, o modelo final. O modelo final pode depois ser usado para prever a variável de saída (Y).
 - 6.1) Considera a estrutura do *StackingClassifier* apresentada nos slides seguintes.
 - 6.2) Valida a tua implementação seguindo o protocolo:
 1. Usa o dataset *breast-bin.csv*
 2. Usa o *sklearn.preprocessing.StandardScaler* para standardizar os dataset.
`breast_dataset.X = StandardScaler().fit_transform(breast_dataset.X)`
 3. Divide o dataset em treino e teste
 4. Cria o modelo *KNNClassifier*
 5. Cria o modelo *LogisticRegression*
 6. Cria um segundo modelo *KNNClassifier* (*modelo final*)
 7. Cria o modelo *StackingClassifier* usando os classificadores anteriores. O segundo modelo *KNNClassifier* deve ser usado como modelo final.
 8. Treina o modelo. Qual o score obtido?

Objeto *StackingClassifier*

- No sub-package *ensemble*, adiciona o modulo *stacking_classifier.py* com o objeto *StackingClassifier*.
- *class StackingClassifier*:
 - Parâmetros:
 - *models* – conjunto de modelos
 - *final_model* – o modelo final
 - Métodos:
 - *fit* – treina os modelos do *ensemble*
 - *predict* – estima a variável de saída usando os modelos treinados e o modelo final
 - *score* – calcula o erro entre as previsões e os valores reais

Objeto *StackingClassifier*

- *StackingClassifier.fit*:

1. Treina o conjunto de modelos
2. Obtém previsões de cada modelo treinado anteriormente
3. Treina o modelo final usando as previsões obtidas anteriormente
4. Retorna ele próprio

- *StackingClassifier.predict*:

1. Obtém previsões de cada modelo no conjunto de modelos
2. Obtém as previsões finais usando o modelo final e as previsões obtidas anteriormente

Objeto *StackingClassifier*

- *StackingClassifier.score*:

1. Estima os valores de Y usando os modelos treinados e o modelo final
2. Calcula a *accuracy* entre os valores reais e as previsões