

Aprendizado de Máquina

Ensembles

André C. P. L. F. de Carvalho
ICMC-USP



Combinação de classificadores

- Condições necessárias para um bom desempenho
 - Diversidade
 - Classificadores base devem ser independentes
 - Ideal: cometer erros diferentes
 - Acurácia
 - Desempenho dos classificadores base deve ser melhor que classificação aleatória
 - Classe majoritária

07/06/2019

André de Carvalho - ICMC/USP

4

Principais tópicos

- Ensembles
- Combinação de classificadores
 - Viés e variância
 - Boosting
 - Bagging
 - Stacking
 - Ensembles de árvores

07/06/2019

André de Carvalho - ICMC/USP

2

Exemplo

- Sejam 3 classificadores induzidos para os mesmos dados, com acurácia 0.6
 - Se eles cometem os mesmos erros
 - Acurácia do ensemble será 0.6
 - Se eles são completamente independentes
 - Ensemble erra classificação apenas se pelo menos 2 classificadores erram na predição

$$erro_{ens} = \sum_{i=2}^3 \binom{3}{i} e^i (1-e)^{3-i} \quad e: \text{ taxa de erro } (0.35 < 0.4)$$

Distribuição Binomial

07/06/2019

André de Carvalho - ICMC/USP

5

Ensembles

- Procuram melhorar acurácia combinando predições de múltiplos estimadores
 - Classificação
 - Constroem conjunto de classificadores a partir de dados de treinamento
 - Classificadores base
 - Classe do novo exemplo é definida pela agregação da predição dos múltiplos classificadores base
 - Também podem ser usados em tarefas de regressão e de agrupamento de dados

07/06/2019

André de Carvalho - ICMC/USP

3

Exercício

- Sejam agora cinco classificadores
 - Predições iguais
 - Taxa de erro = 0.4
 - Predições independentes
 - Taxa de erro = ?

07/06/2019

André de Carvalho - ICMC/USP

6

Estruturas de combinação

- **Combinação Paralela**
 - Treinamento independente
 - Algoritmos aplicados a:
 - Mesmo conjunto dados
 - Conjuntos de dados formados por diferentes amostras do conjunto de dados original
 - Conjuntos de dados com diferentes atributos preditivos do conjunto de dados originais
 - Explora semelhanças e diferenças

07/06/2019 André de Carvalho - ICMC/USP 7

Estruturas de combinação

- **Combinação hierárquica**
 - Mistura das combinações anteriores
 - Caso especial: stacking

07/06/2019 André de Carvalho - ICMC/USP 10

Estruturas de combinação

- **Combinação Paralela**

07/06/2019 André de Carvalho - ICMC/USP 8

Combinações paralelas

- Pode ocorrer pela manipulação de:
 - Conjunto de treinamento
 - Bosting e Baging
 - Atributos preditivos
 - Ensemble de árvores
 - Rótulos das classes
 - Multiclasses e multirrótulo
 - Algoritmo de aprendizado
 - Modelos gerados por algoritmo(s)

07/06/2019 André de Carvalho - ICMC/USP 11

Estruturas de combinação

- **Combinação em cascata (sequencial)**
 - Saída de um classificador é utilizada como entrada para o próximo classificador
 - Não precisa combinar saídas
 - Problema: propagação de erro

07/06/2019 André de Carvalho - ICMC/USP 9

Combinação de previsões

- **Combinação de previsões**
 - Voto (média)
 - Voto (média) ponderado
 - Algoritmo combinador (*stacking*)

07/06/2019 André de Carvalho - ICMC/USP 12

Decomposição viés-variância

- Em geral
 - Quanto mais forte a suposição de um classificador sobre o espaço de decisão, maior seu viés
 - Ex. Árvore podada faz suposição mais forte, por basear a fronteira em menos atributos
 - Menos consistente com os dados de treinamento
 - Tem maior viés (e menor variância)

07/06/2019 André de Carvalho - ICMC/USP 13

Bagging (Bootstrap Agregating)

- Indicado para classificadores instáveis
 - Pequena mudança nos dados de treinamento afeta modelo de classificação induzido
 - Redes neurais e árvores de decisão
 - Por que eles são instáveis?
- Não são indicados para classificadores estáveis
 - Erro geralmente causado por viés do classificador base
- Menos sensível a *overfitting* quando dados têm ruído

07/06/2019 André de Carvalho - ICMC/USP 16

Decomposição viés-variância

- Em geral
 - Algoritmo de classificação gera modelos diferentes para mesmo conjunto de dados
 - Variabilidade do conjunto de treinamento leva a variância nos erros de predição
- Erro de modelo é definido por três componentes
 - Viés + variância + ruído

07/06/2019 André de Carvalho - ICMC/USP 14

Bagging

- Seja o conjunto de dados de treinamento $\{x_1, x_2, x_3, x_4, x_5, x_6\}$

$x_1, x_6, x_3, x_5, x_3, x_1$	Amostra 1
$x_3, x_4, x_1, x_5, x_5, x_1$	Amostra 2
...	...
$x_1, x_2, x_4, x_2, x_6, x_4$	Amostra m

07/06/2019 André de Carvalho - ICMC/USP 17

Bagging (Bootstrap Agregating)

- Cada classificador é induzido por uma amostra diferente do conjunto de treinamento
 - Mesmo tamanho do conjunto original
 - Usa *bootstrap*
- Classe definida por votação
- Tende a reduzir variância associada com classificadores base

07/06/2019 André de Carvalho - ICMC/USP 15

Bagging

```

    graph LR
      A1["x1, x6, x3, x5, x3, x1"] --> C1["Classificador 1"]
      A2["x3, x4, x1, x5, x5, x1"] --> C2["Classificador 2"]
      Dots["..."]
      Am["x1, x2, x4, x2, x6, x4"] --> Cm["Classificador m"]
      C1 --> E["Estimador"]
      C2 --> E
      Cm --> E
    
```

07/06/2019 André de Carvalho - ICMC/USP 18


Boosting

- Conjunto de técnicas
 - Adaboost é uma das mais conhecidas
- A cada iteração
 - Induz classificador
 - Pondera cada exemplo do conjunto de dados completo pelo desempenho do classificador base
 - Quanto mais difícil de ser aprendido, maior o peso associado ao exemplo
 - Maior probabilidade de ser escolhido na próxima iteração
- Boosting funciona de forma semelhante a minimização por gradiente descendente

07/06/2019 André de Carvalho - ICMC/USP 19

Stacking

- Um algoritmo estimador aprende a combinar predições de modelos base
 - Modelos gerados por algoritmos base
 - Saídas combinadas por algoritmo estimador
 - Algoritmo de AM
- Algoritmos base podem ser:
 - Homogêneos
 - Heterógenos



07/06/2019 André de Carvalho - ICMC/USP 22

Boosting

- Seja o conj. treinamento $\{x_1, x_2, x_3, x_4, x_5\}$

Exemplos:	x_1	x_2	x_3	x_4	x_5	Soma dos pesos = 1.0
Pesos atuais:	0.2	0.2	0.2	0.2	0.2	
Classificação:	C	I	C	C	I	C: correta I: incorreta
Novos pesos:	0.2	0.4	0.2	0.2	0.4	

Exemplos:	x_1	x_2	x_3	x_4	x_5
Pesos atuais:	0.2	0.4	0.2	0.2	0.4
Classificação:	C	I	C	I	C
Novos pesos:	0.2	0.6	0.2	0.4	0.4

...

07/06/2019 André de Carvalho - ICMC/USP 20

Ensembles de ADs

- Combina a predição de várias árvores de decisão
- Duas principais abordagens:
 - Random forests
 - Extreme Gradient Boosting
 - Algoritmo CART ou baseado no CART

07/06/2019 André de Carvalho - ICMC/USP 23

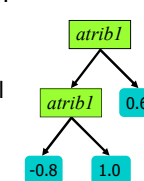
Boosting

- Indicado para classificadores base fracos
 - Acurácia ligeiramente melhor que palpite aleatório
- Convergência rápida
- Pouco indicado para dados com ruídos e pequenos conjuntos de dados
 - Por focar em exemplos difíceis de serem classificados

07/06/2019 André de Carvalho - ICMC/USP 21

Algoritmo CART

- Classification and regression trees
 - Árvore binária
 - Nó folha possui um valor real
 - Que pode ser usado para regressão ou classificação



07/06/2019 André de Carvalho - ICMC/USP 24

Random Forests (RFs)

- Combinar ADs, mas pode usar modelos gerados por qualquer algoritmo de AM
- Combina k ADs
 - Cada árvore é induzida usando um subconjunto aleatório dos atributos preditivos
 - Usado na escolha do atributo para cada nó
 - Hiper-parâmetros definem número de ADs e número de atributos preditivos para cada AD
 - Classificação ocorre por voto majoritário

07/06/2019 André de Carvalho - ICMC/USP 25

Random Forests (RFs)

- Forest-RC (*Random Combination*)
 - Expande número de atributos criando combinações lineares aleatórias de atributos
 - A cada nó, F combinações de L atributos são aleatoriamente geradas
 - Combina atributos utilizando pesos aleatoriamente gerados entre -1 e +1
 - Cada combinação é um novo atributo
 - Usada quando conjunto de dados tem poucos atributos preditivos

07/06/2019 André de Carvalho - ICMC/USP 28

Random Forests (RFs)

- *Bagging* pode ser visto como um caso especial de RFs
 - N é número de atributos do conjunto de dados
 - RFs usa *bootstrap* de forma similar a *bagging* para selecionar exemplos de treinamento
- Várias alternativas para escolher aleatoriamente os atributos preditivos
 - Forest-RI (Random Input Selection)
 - Forest-RC (Random Combination)

07/06/2019 André de Carvalho - ICMC/USP 26

Extreme Gradient Boosting

- XGBoost
- Combina árvores geradas pelo algoritmo CART
- Treinamento aditivo
 - Induz uma árvore
 - Inclui ela no ensemble
 - Induz próxima árvore
 - ...
- Pondera a resposta de cada árvore para reduzir complexidade do modelo final

07/06/2019 André de Carvalho - ICMC/USP 29

Random Forests (RFs)

- Forest – RI (*Random Input Selection*)
 - Seleciona aleatoriamente, para cada nó, um subconjunto de F atributos preditivos
 - Algoritmo CART é usado para crescer as árvores sem poda
 - Problema: conjunto de dados com poucos atributos preditivos
 - Pode selecionar atributos fortemente correlacionados

07/06/2019 André de Carvalho - ICMC/USP 27

Conclusão

- Combinação de estimadores em geral aumenta desempenho preditivo
 - E reduz variância
- As vezes chamado de meta-aprendizado
- Regressão
 - Média simples ou ponderada

07/06/2019 André de Carvalho - ICMC/USP 30


Exercício

- Seja o seguinte cadastro de pacientes:

Nome	Febre	Enjôo	Manchas	Dores	Diagnóstico
João	sim	sim	pequenas	sim	doente
Pedro	não	não	grandes	não	saudável
Maria	sim	sim	pequenas	não	saudável
José	sim	não	grandes	sim	doente
Ana	sim	não	pequenas	sim	saudável
Leila	não	não	grandes	sim	doente

Andre Ponce de Leon de Carvalho 31

Perguntas



André C P L F de Carvalho 34

Exercício

- Avaliar o desempenho de bagging, boosting, stacking e random forest
 - Usar C4.5 como classificador base e combinador
- Como seriam classificados os exemplos?
 - (Luis, não, não, pequenas, sim)
 - (Laura, sim, sim, grandes, sim)

Andre Ponce de Leon de Carvalho 32

Tarefa

- Avaliar o desempenho de bagging, boosting, stacking, random forest e XGBoost
 - Usar C4.5 como classificador base e combinador
 - Comparar com algoritmo C4.5 usado de forma isolada
- Usar conjunto de dados glass da UCI
- Particionar os dados utilizando 10-fold cross-validation

07/06/2019 André de Carvalho - ICMC/USP 33