

Avaliação de Modelos (parte 2)

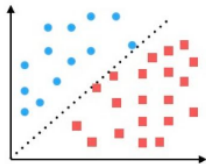
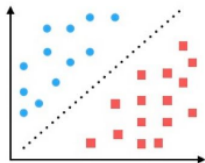
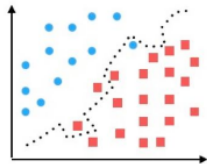
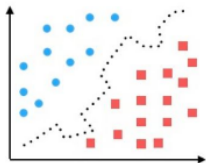
Prof. Jefferson T. Oliva

Aprendizado de Máquina e Reconhecimento de Padrões (AM28CP)
Engenharia de Computação
Departamento Acadêmico de Informática (Dainf)
Universidade Tecnológica Federal do Paraná (UTFPR)
Campus Pato Branco

- Método *Holdout*
- Validação Cruzada
- Medidas de Avaliação de Modelos

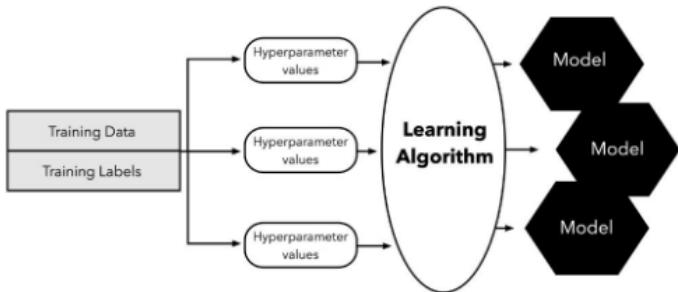
Introdução

- É desejável estimar o desempenho da generalização do modelo
 - Desempenho preditivo para dados não vistos

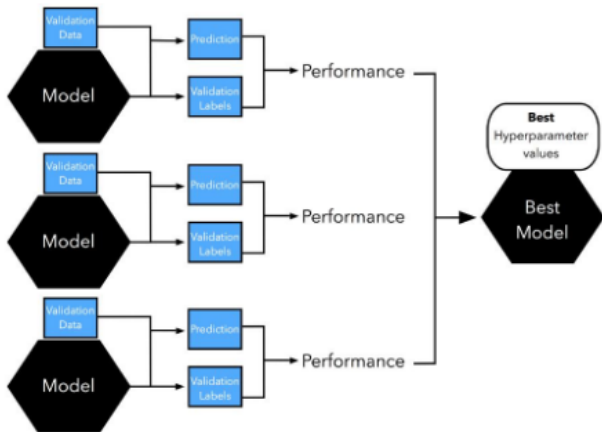


Introdução

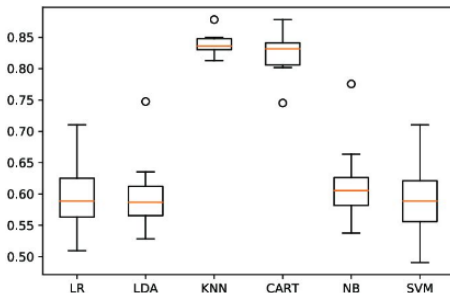
- É desejável aumentar o desempenho preditivo ajustando o algoritmo de aprendizagem e selecionando o modelo de melhor desempenho de um determinado espaço de hipóteses



- É desejável aumentar o desempenho preditivo ajustando o algoritmo de aprendizagem e selecionando o modelo de melhor desempenho de um determinado espaço de hipóteses

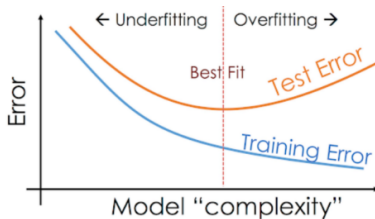


- O objetivo pode ser também identificar o algoritmo de ML mais adequado para o problema em questão
 - Nesse caso, vários algoritmos de aprendizado de máquina são comparados



Introdução

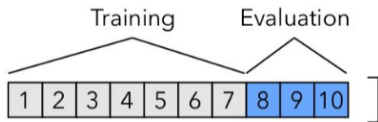
- O erro do conjunto de treinamento é um estimador com viés otimista do erro de generalização
- O erro do conjunto de teste é um estimador sem viés do erro de generalização



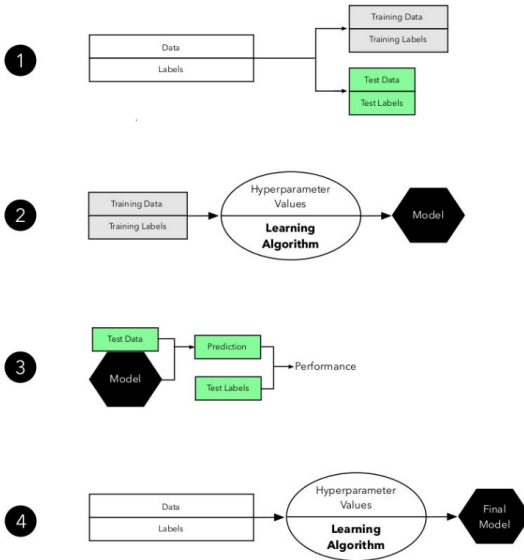
Método *Houldout*

Método *Houldout*

- Esse método utiliza uma parte (e.g. 2/3) do conjunto de dados para o treinamento do modelo e o restante, para teste
- Muitas vezes, usar o *houdout* não é uma boa ideia para a avaliação de modelos
- Os exemplos podem não ser representativos
 - Por exemplo, pode faltar exemplos de uma classe ou haver desbalanceamento
 - Para esse caso, uma solução seria a estratificação das partições

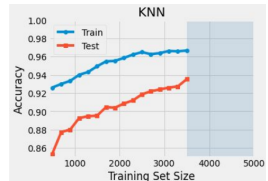
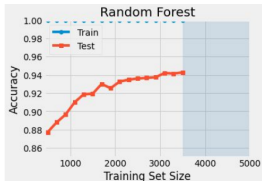
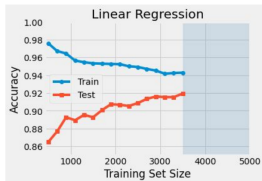


Método *Houldout*

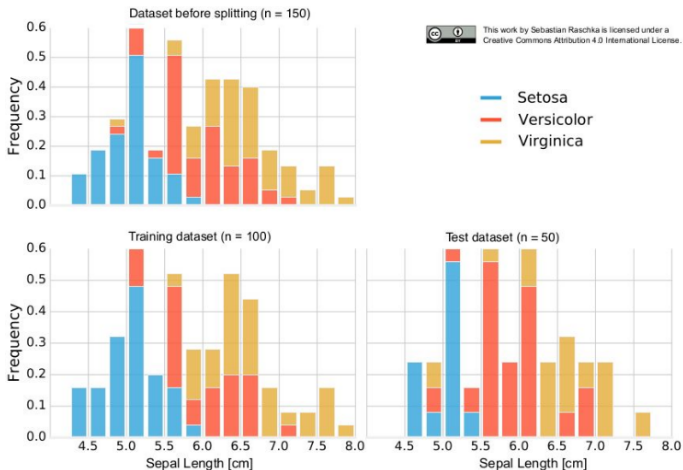


Método *Houldout*

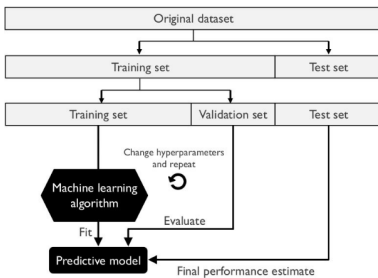
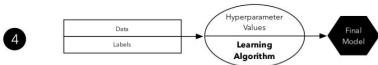
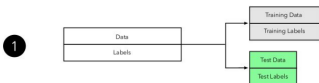
- O erro obtido utilizando o conjunto de teste no modelo preditivo é pessimista
- Mas a variação nos dados de treinamento não é levada em conta



- Problemas com subamostragem (violação de independência)



- Método *holdout*: seleção de modelos vs. avaliação de modelos



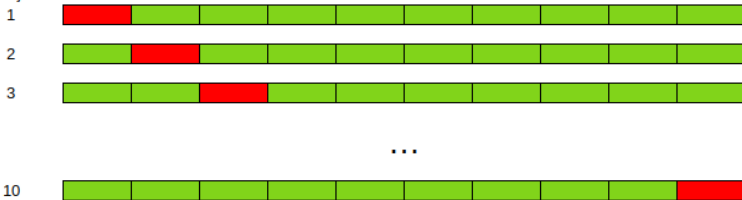
Validação Cruzada

- A divisão dos dados entre conjuntos de treino e teste uma única vez pode acarretar em vícios
- E se fazermos a divisão de dados mais de uma vez?
 - A ocorrência de anormalidade nos dados fica diluída
- A validação cruzada é um método de reamostragem para a geração de diversas amostras aleatória a partir da mesma população
 - Em outras palavras, diversos conjuntos de treino e de teste diferentes são gerados
- Exemplos de abordagens de validação cruzada:
 - *k-fold*
 - *leave-me-out*

Validação Cruzada

- O *k-fold*, por exemplo, divide o conjunto de dados em k conjuntos, sendo um para teste e os $k - 1$ restantes para treinamento do modelo em um processo que é repetido k vezes
 - Os valores de k mais comum são 5 e 10

Iteração



Legenda:



Conjunto de teste



Conjuntos de treinamento

- *Leave-me-out* é um caso específico de *k-fold*, onde *k* é igual ao a quantidade total de exemplos
 - Por mais que apresente uma avaliação completa (obtenção do erro verdadeiro) sobre a variação do modelo, esta abordagem possui custo computacional elevado
- Recomendo apenas em situações onde poucos dados estão disponíveis
- Por outro lado, a validação cruzada *10-fold* é a mais recomendada, pois consiste em uma avaliação robusta
 - Além do equilíbrio entre viés e variância, resulta na aproximação do erro verdadeiro
 - Recomendado para pequenas e médias bases de dados
- A validação cruzada *5-fold* é usada caso a base de dados seja considerada grande

- Problema: desbalanceamento por classe na amostragens
 - Solução: estratificação, onde os exemplos são mantidos proporcionalmente em relação às classes
 - No entanto, estratificação não é possível no *leave-me-out*

Medidas de Avaliação de Modelos

- Matriz de Confusão

- Verdadeiro positivo (V_P): quantidade de exemplos normais (positivos) classificados como positivo ou normal
- Verdadeiro negativo (V_N): quantidade de exemplos negativos (novidades) classificados como negativo ou novidade
- Falso positivo (F_P): quantidade de exemplos negativos classificados como normal
- Falso negativo (F_N): quantidade de exemplos normais classificados como novidade

	<i>Positivo</i>	<i>Negativo</i>	Total
<i>Positivo</i>	V_P	F_N	$V_P + F_N$
<i>Negativo</i>	F_P	V_N	$F_P + V_N$
Total	$V_P + F_P$	$F_N + V_N$	$V_P + F_N + F_P + V_N$

- Acurácia

$$Acc = \frac{VP + VN}{VP + FN + VN + FP}$$

- Sensibilidade (revocação – *recall*)

$$Sen = \frac{VP}{VP + FN}$$

- Especificidade

$$Esp = \frac{VN}{VN + FP}$$

- Valor preditivo positivo (precisão)

$$VPP = \frac{VP}{VP + FP}$$

- Valor preditivo negativo

$$VPN = \frac{VN}{VN + FN}$$

- f1-score

$$VPN = 2 * \frac{VPP * Sen}{VPP * Sen} \rightarrow VPN = 2 * \frac{\text{precisão} * \text{revocação}}{\text{precisão} + \text{revocação}}$$

Medidas de Avaliação de Modelos

- As medidas baseadas na matriz de confusão foram propostas para a avaliação de modelos de classificação
- Os exemplos de medidas apresentados, por mais que sejam no contexto de modelos binários, algumas delas podem ser adaptadas para a classificação multiclasse, tais como: acurácia, revocação, sensibilidade e *f1-score*
 - Especificidade é a "revocação" para a classe negativa
 - Valor preditivo negativo é a "precisão" para a classe negativa
- Na aplicação da validação cruzada, cada uma das medidas de avaliação pode ser calculada para cada *fold* de teste
 - Geralmente são obtidos os valores de média e de desvio-padrão para cada medida

Medidas de Avaliação de Modelos

- Para a regressão, as medidas são geralmente calculadas com base na diferença entre o valor real (*alvo/target*) e o predito

- Exemplos de medidas:

- Erro quadrático médio (EQM)

$$EQM = \frac{1}{N} \sum_{i=1}^N (Y_i - \bar{Y}_i)^2$$

- Raiz quadrada do EQM (*root-mean-square error* – RMSE)

$$RMSE = \sqrt{EQM}$$

- Erro médio absoluto (EMA)

$$EMA = \frac{1}{N} \sum_{i=1}^N |Y_i - \bar{Y}_i|$$

onde Y_i é o valor real do i -ésimo exemplo, \bar{Y}_i é o i -ésimo resultado da predição e N é a quantidade de exemplos

- Exemplos de medidas (regressão):
 - Coeficiente de determinação (R^2)

$$R^2 = 1 - \frac{\sum_{i=1}^N (Y_i - \bar{Y})^2}{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}$$

onde \hat{Y}_i é o valor médio dos valores reais

- Erro percentual absoluto médio (*mean absolute percentual error* – MAPE)

$$MAPE = \frac{1}{n} \sum_{i=1}^N \frac{|Y_i - \bar{Y}_i|}{Y_i}$$

- Assim como na classificação, na regressão também pode ser aplicada a validação cruzada
 - Obtenção de valores médios de EQM, RMSE, EMA, R^2 ...
- As avaliações podem ser complementadas por
 - Intervalo de confiança
 - Testes estatísticos de hipótese (comparação de modelos)

Medidas de Avaliação de Modelos

Intervalo de confiança

- Indica, com um determinado nível de segurança (geralmente 95%), onde se espera que esteja o valor verdadeiro de uma medida de desempenho de um modelo
- O intervalo de confiança fornece uma faixa provável de valores
 - Expressa a incerteza da estimativa
 - Exemplo: a acurácia verdadeira está entre 85% e 91%, considerando 95% de confiança
- Mostra que a medida de avaliação pode variar de acordo com a amostra de dados utilizada

Medidas de Avaliação de Modelos

Intervalo de confiança

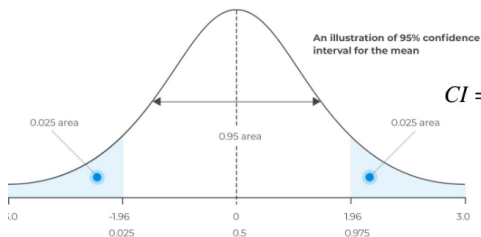
- Dado um valor de erro (ERR), o intervalo de confiança pode ser determinado pela seguinte equação

$$IC = ERR \pm z \sqrt{\frac{ERR(1 - ERR)}{n}}$$

- A constante z para os seguintes intervalos de confiança:
 - 99% $\rightarrow z = 2,58$
 - 95% $\rightarrow z = 1,96$
 - 90% $\rightarrow z = 1,64$

Medidas de Avaliação de Modelos

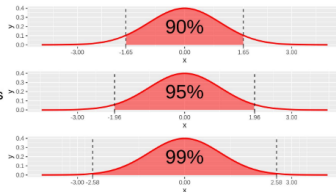
Intervalo de confiança



$$CI = ERR_S \pm z \sqrt{\frac{ERR_S(1 - ERR_S)}{n}}$$

The z constant for different confidence intervals

- 99%: $z=2.58$
- 95%: $z=1.96$
- 90%: $z=1.64$



Medidas de Avaliação de Modelos

Testes estatísticos de hipótese

- Podem ser utilizados para a comparação entre dois ou mais classificadores verificar se há diferença estatística entre eles
- O teste estatístico de hipótese determina a probabilidade que uma diferença observada de forma empírica seja de fato somente ao acaso
- Esse tipo de teste determina a probabilidade da hipótese nula, de que as duas amostras vieram da mesma distribuição
- Para isso, é considerado um nível de significância, comumente (5%)
 - Por convenção, hipótese nula é rejeitada e diz-se que a diferença é estatisticamente significativa se a probabilidade da hipótese nula for menor que 5%, ou seja, $p < 0,05$

Medidas de Avaliação de Modelos

Testes estatísticos de hipótese

- Para a escolha do teste apropriado, duas principais observações em relação às características dos dados devem ser realizadas:
 - Se os dados são pareados ou não-pareados
 - Se os dados estão dentro da distribuição normal, o que determina se o teste deve ser paramétrico ou não-paramétrico
- Exemplos de testes estatísticos de hipótese:
 - Paramétrico para dados pareados: t de Student e *two-way* ANOVA (*Analysis of Variance* – análise de variância)
 - Não-paramétrico para dados pareados: Wilcoxon e Friedman
 - Paramétrico para dados não pareados: t de Student não pareado e *one-way* ANOVA
 - Não-paramétrico para dados não pareados: Mann-Whitney e Kruskal-Wallis

- Caso mais de dois classificadores sejam comparados e uma diferença estatística seja constatada, um pós teste (*post hoc*) deve ser aplicado
 - A escolha de um teste *post hoc* segue os mesmos critérios da definição do teste estatístico de hipótese adequado
 - Por exemplo: o pós-teste de Nemenyi pode ser executado após o teste de Friedman, caso neste último tenha sido observada uma diferença estatisticamente significativa



CASANOVA, D.

Model evaluation 2. Aprendizado de Máquina.

Slides. Engenharia de Computação. Dainf/UTFPR, 2020.



DOMINGOS, Pedro. A unified bias-variance decomposition.

In: Proceedings of 17th international conference on machine learning.

Morgan Kaufmann Stanford, 2000. p. 231-238.



RASCHKA, S.; MIRJALILI, V.

Python Machine Learning.

Packt, 2017.



ZADROZNY, B. *Avaliação experimental. Aprendizado de Máquina.*

Slides. Ciência da Computação. UFF, 2010.