

Mineração de Dados

Avaliação de Modelos



- 1 Avaliação e Seleção de Modelos de Classificação
- 2 Métodos Auxiliares
- 3 Desbalanceamento dos Dados
- 4 Avaliação de Modelos: scikit-learn

Avaliação e Seleção de Modelos de Classificação

Introdução

- ▶ Etapas do desenvolvimento do modelo
 - ▶ Treinamento ou geração de modelos
 - ▶ Seleção de modelos
 - ▶ Avaliação e análise de modelos
- ▶ Comparação
 - ▶ Entre modelos iguais: quais parâmetros são melhores?
 - ▶ Entre os diferentes modelos: qual o melhor modelo?
- ▶ Métricas de avaliação
- ▶ Comparando classificadores
 - ▶ Acurácia/Erro
 - ▶ Intervalos de confiança/análise estatística
 - ▶ Curva ROC

Introdução

- ▶ É apropriado utilizar um conjunto de validação/teste para avaliar os modelos candidatos
 - ▶ Não deve-se utilizar os dados de treinamento
- ▶ Métodos para apoiar a avaliação dos modelos
 - ▶ Holdout
 - ▶ Validação cruzada
 - ▶ Bootstrap

Acurácia/Erro

- ▶ Avaliar a capacidade do modelo em prever o valor da supervisão sobre um novo conjunto de dados
- ▶ **Dados diferentes daqueles usados para a geração dos modelos!**
- ▶ Métodos auxiliares de avaliação
 - ▶ Holdout
 - ▶ Validação cruzada
 - ▶ Bootstrap

Acurácia/Erro

- ▶ Define-se as instâncias marcadas como sendo da **classe de interesse** como **positivas**
- ▶ As tuplas da outra classe são ditas **negativas**
- ▶ Por exemplo
 - ▶ *buys-computer = yes*: Positiva
 - ▶ *buys-computer = no*: Negativa
- ▶ Com essa definição e um modelo de classificação, 4 possibilidades podem ocorrer ao avaliar o modelo sobre novos dados

Matriz de Confusão

- ▶ Verdadeiros-positivos (TP): Instâncias positivas e que são classificadas (corretamente) pelo modelo como positivas
- ▶ Verdadeiros-negativos (TN): Instâncias negativas e que são classificadas (corretamente) pelo modelo como negativas
- ▶ Falsos-positivos (FP): Instâncias que são negativas mas que são classificadas (incorretamente) pelo modelo como positivas
- ▶ Falsos-negativos (FN): Instâncias que são positivas mas que são classificadas (incorretamente) pelo modelo como negativas

		Predicted class		
		<i>yes</i>	<i>no</i>	Total
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total		<i>P'</i>	<i>N'</i>	<i>P + N</i>

Matriz de Confusão

► Matriz de confusão

		Predicted class		
		<i>yes</i>	<i>no</i>	Total
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total		<i>P'</i>	<i>N'</i>	<i>P + N</i>

► Exemplo:

Actual class \ Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	6954	46	7000
buy_computer = no	412	2588	3000
Total	7366	2634	10000

Matriz de Confusão

- ▶ A acurácia (classificação correta) pode ser contabilizada usando a diagonal principal
- ▶ Deseja-se que se tenha pequenas quantidades “falsas”
- ▶ No caso multiclasse, MC_{ij} indica a quantidade de instâncias da classe i que foi classificada como sendo da classe j
- ▶ A matriz de confusão pode conter linhas e colunas adicionais para contagem de valores totais

Acurácia/Erro

		Predicted class		
		<i>yes</i>	<i>no</i>	Total
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total		<i>P'</i>	<i>N'</i>	<i>P + N</i>

- ▶ Acurácia: porcentagem das instâncias classificadas corretamente

- ▶
$$\frac{TP + TN}{P + N}$$

- ▶ Porcentagem de Erro: 1 – acurácia

- ▶
$$1 - \frac{TP + TN}{P + N} = \frac{FP + FN}{P + N}$$

Problema de Desbalanceamento dos Dados

- ▶ Uma das classes pode ser rara
 - ▶ Fraude
 - ▶ Indivíduo com uma dada doença

		Predicted class		
		<i>yes</i>	<i>no</i>	Total
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total		<i>P'</i>	<i>N'</i>	<i>P + N</i>

- ▶ Sensitividade: porcentagem das instâncias positivas classificadas corretamente
 - ▶ $\frac{TP}{P}$
- ▶ Especificidade: porcentagem das classes negativas classificadas corretamente
 - ▶ $\frac{TN}{N}$

Outras Medidas Comuns

		Predicted class		
		<i>yes</i>	<i>no</i>	
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
	Total	<i>P'</i>	<i>N'</i>	<i>P + N</i>

- Precisão: porcentagem das instâncias classificadas como positivas que são realmente positivas

$$\frac{TP}{TP + FP}$$

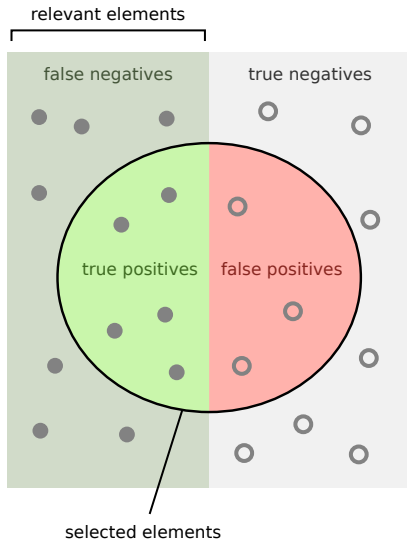
- Indica a exatidão do modelo

- Abrangência (*Recall*): porcentagem das classes positivas classificadas corretamente


$$\frac{TP}{P}$$

- Indica a completude do modelo
- Igual à sensibilidade


Precisão e Abrangência



How many selected items are relevant?

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$


How many relevant items are selected?

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$


Outras Medidas Comuns

► Medida-F: *F-measure*, F_1 ou *F-score*

- $$F = \frac{2 \times \text{Precisão} \times \text{Abrangência}}{\text{Precisão} + \text{Abrangência}}$$
- Média harmônica entre precisão e abrangência

► F_β

- $$F = \frac{(1 - \beta^2) \times \text{Precisão} \times \text{Abrangência}}{\beta^2 \text{Precisão} + \text{Abrangência}}$$
- F_2 é uma medida comum e que enfatiza abrangência sobre precisão
- $F_{0.5}$ é uma medida comum e que enfatiza precisão sobre a abrangência

Avaliação

<i>Measure</i>	<i>Formula</i>
accuracy, recognition rate	$\frac{TP + TN}{P + N}$
error rate, misclassification rate	$\frac{FP + FN}{P + N}$
sensitivity, true positive rate, recall	$\frac{TP}{P}$
specificity, true negative rate	$\frac{TN}{N}$
precision	$\frac{TP}{TP + FP}$
F , F_1 , F -score, harmonic mean of precision and recall	$\frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$
F_β , where β is a non-negative real number	$\frac{(1 + \beta^2) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}}$

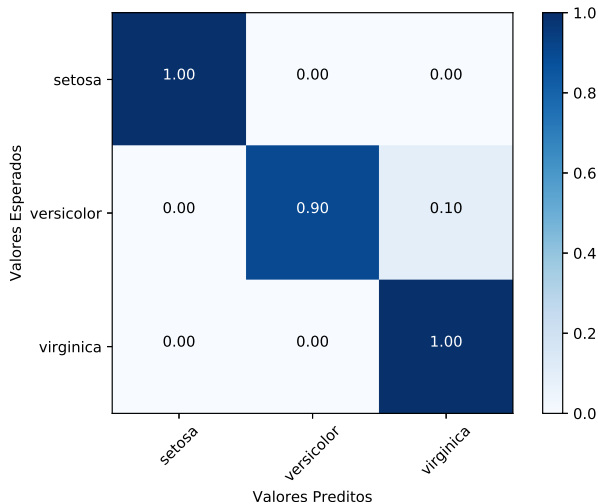
Exemplo

Actual Class\Predicted class	cancer = yes	cancer = no	Total	Recognition(%)
cancer = yes	90	210	300	30.00 (<i>sensitivity</i>)
cancer = no	140	9560	9700	98.56 (<i>specificity</i>)
Total	230	9770	10000	96.40 (<i>accuracy</i>)

■ $Precision = 90/230 = 39.13\%$

$$Recall = 90/300 = 30.00\%$$

Matriz de Confusão



Avaliação de Modelos

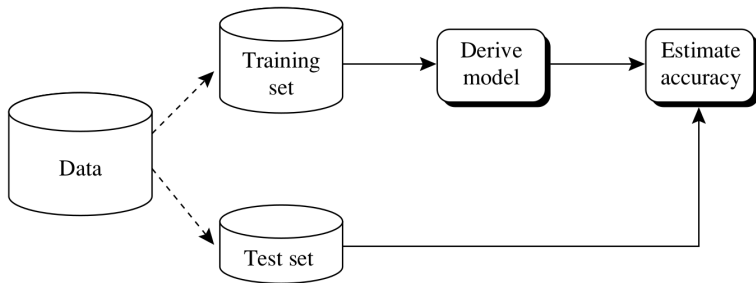
- ▶ Essas formas de avaliação são suficientes?
- ▶ As instâncias podem não ser classificadas unicamente
 - ▶ Neste caso, é mais apropriado que se tenha uma distribuição de probabilidade
- ▶ As instâncias podem não ter o mesmo peso na avaliação
 - ▶ ou erros diferentes podem ter pesos diferentes
- ▶ Velocidade
- ▶ Robustez: o modelo é acurado sobre dados com ruído ou com valores ausentes?
- ▶ Escalabilidade
- ▶ Interpretabilidade

Métodos Auxiliares

Holdout

► Holdout

- Os dados são divididos em 2 conjuntos independentes
- Dados de treinamento: construção do modelo (por exemplo, 2/3)
- Dados de teste: estimacão da acurácia (por exemplo, 1/3)
- Pode-se repetir o *holdout* k vezes e a acurácia pode ser calculada pela média das acurácias obtidas

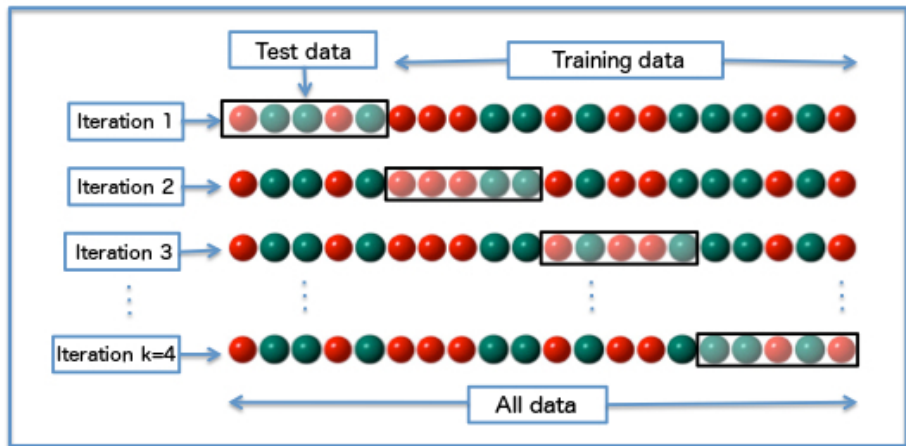


Validação Cruzada

▶ Validação Cruzada

- ▶ Particiona os dados em k mutuamente exclusivos conjuntos
- ▶ Os k conjuntos tem aproximadamente o mesmo número de elementos
- ▶ Na i -ésima iteração, utilize D_i como conjunto de teste e os demais conjuntos como dados de treinamento
- ▶ k -fold com $k = 10$ é o mais popular
- ▶ O menor conjunto de dados resulta no *leave-one-out*
- ▶ A versão que busca manter as proporções dos dados é chamada de Validação Cruzada Estratificada
- ▶ Sugere-se a utilização do *stratified 10-fold cross-validation*

Validação Cruzada



Bootstrap

- ▶ Funciona bem com poucos dados
- ▶ Amostra-se os dados de treinamento uniformemente e com reposição
- ▶ Existem diversas variantes e a mais comum é o *bootstrap* .632
 - ▶ Um conjunto de dados com m tuplas é amostrado m vezes (com reposição), resultando em um conjunto de treinamento de m elementos
 - ▶ As instâncias que não foram selecionadas para o treinamento irão compor o conjunto de teste
 - ▶ Aproximadamente 63.2% dos dados fazem parte dos dados de treinamento, e os demais 36.8% do conjunto de teste
 - ▶ A chance de uma instância ser sorteada ao acaso é $1/m$, então $(1 - 1/m)^m \approx e^{-1} = 0.368$ é a chance total dela não ser sorteada
 - ▶ O procedimento é repetido por k vezes
 - ▶ A qualidade do modelo é então definida como:

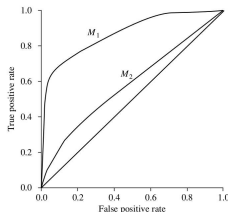
$$Q_{bootstrap}(D) = \frac{1}{k} \sum_{i=1}^k [0.632 \times \text{Acur}(D_{i,\text{teste}}) + 0.368 \times \text{Acur}(D_{i,\text{treinamento}})]$$

Significância Estatística

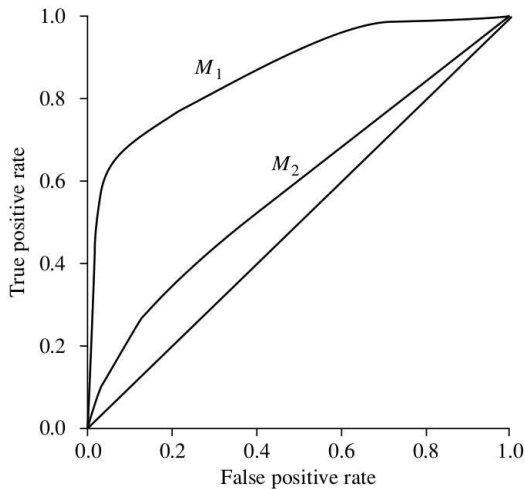
- ▶ Os k testes realizados via Validação Cruzada (por exemplo) possuem variância
- ▶ A diferença média observada pode não ter significância
- ▶ Uma forma de avaliar a significância é
 - ▶ Teste-t
 - ▶ Hipótese nula de que as amostras são similares
 - ▶ Comparação pareada
 - ▶ Tipicamente usa-se 5% de nível de significância
 - ▶ Se o p -valor é menor 0.05 então rejeita-se a hipótese nula
- ▶ Testes não paramétricos também podem ser usados
 - ▶ Teste de Wilcoxon
 - ▶ Teste de Mann–Whitney
 - ▶ Teste de Friedman

Curva ROC

- ▶ Curva ROC: Receiver Operating Characteristics
- ▶ Comparação visual de classificadores
- ▶ Mostra o *trade-off* entre a taxa de verdadeiros positivos e de falsos positivos
 - ▶ O eixo vertical representa a taxa de verdadeiros positivos
 - ▶ O eixo horizontal representa a taxa de falsos positivos
 - ▶ A diagonal principal é um limiar para o caso aleatório
- ▶ A área sob a curva ROC é uma métrica de qualidade do modelo

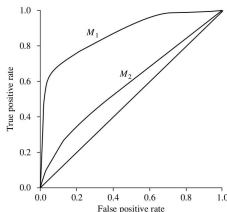


Curva ROC: Exemplo



Curva ROC

- ▶ Construção da Curva ROC
 - ▶ Ordene as tuplas em ordem decrescente da certeza dada pelo modelo da instância pertencer à classe positiva
 - ▶ Tuplas com maior chance de serem classificadas como positivas aparecem no início da lista
 - ▶ Deslocar em relação ao eixo vertical se a classe for realmente positiva
 - ▶ Deslocar em relação ao eixo horizontal se a tupla for um falso positivo
- ▶ Quanto mais próximo da diagonal, menor a qualidade do modelo
- ▶ O melhor modelo é aquele com área sob a curva igual a 1



Curva ROC: Exemplo

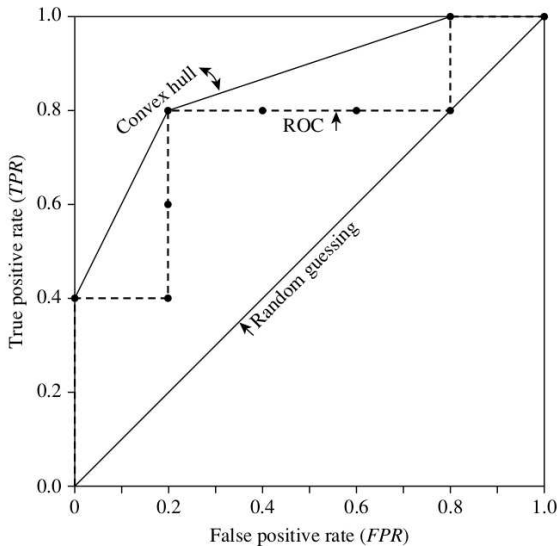
<i>Tuple #</i>	<i>Class</i>	<i>Prob.</i>
1	<i>P</i>	0.90
2	<i>P</i>	0.80
3	<i>N</i>	0.70
4	<i>P</i>	0.60
5	<i>P</i>	0.55
6	<i>N</i>	0.54
7	<i>N</i>	0.53
8	<i>N</i>	0.51
9	<i>P</i>	0.50
10	<i>N</i>	0.40

<i>TP</i>	<i>FP</i>	<i>TN</i>	<i>FN</i>	<i>TPR</i>	<i>FPR</i>
1	0	5	4	0.2	0
2	0	5	3	0.4	0
2	1	4	3	0.4	0.2
3	1	4	2	0.6	0.2
4	1	4	1	0.8	0.2
4	2	3	1	0.8	0.4
4	3	2	1	0.8	0.6
4	4	1	1	0.8	0.8
5	4	0	1	1.0	0.8
5	5	0	0	1.0	1.0

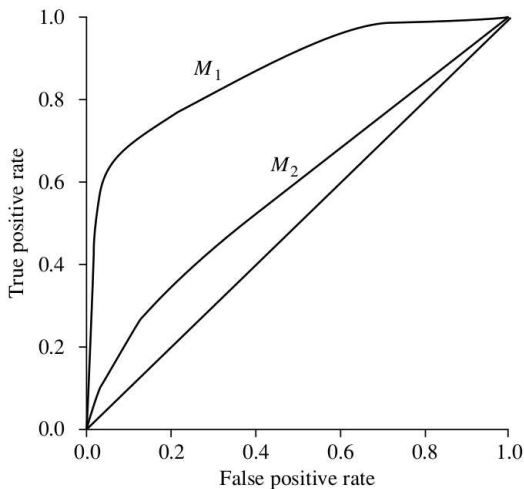
TP: sensibilidade

FP: 1— especificidade

Curva ROC: Exemplo

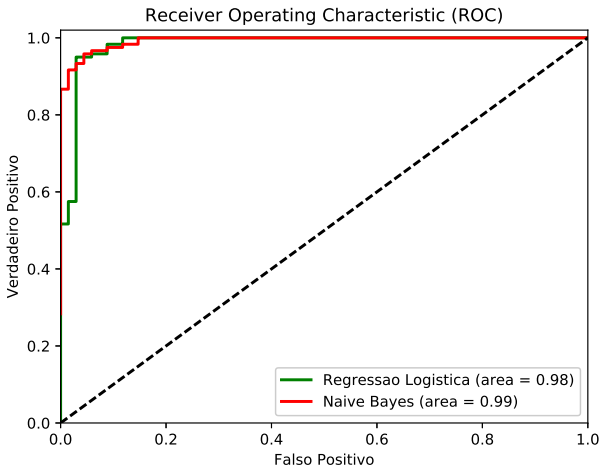


Curva ROC: Exemplo



O modelo M_1 é o melhor segundo essa avaliação.

Curva ROC



Desbalanceamento dos Dados

Desbalanceamento dos Dados

- ▶ Desbalanceamento
 - ▶ Classe majoritária: muitas instâncias
 - ▶ Classe minoritária: poucas instâncias
- ▶ Exemplos
 - ▶ Casos de câncer em bases médicas
 - ▶ Representações de invasores em sistemas
- ▶ Está relacionado ao caso de atribuição de pesos às instâncias
- ▶ Normalmente, os métodos buscam minimizar o erro considerando que os pesos dos falsos positivos e falsos negativos são iguais

Desbalanceamento dos Dados

- ▶ Sensitividade (taxa de verdadeiros positivos) e especificidade (taxa de verdadeiros negativos) ajudam a avaliar a classificação em bases desbalanceadas
- ▶ A curva ROC também ajuda pois avalia a sensibilidade (verdadeiros positivos) e $1 - \text{especificidade}$ (falsos positivos)
- ▶ Alguns meios de tratar o desbalanceamento dos dados
 - ▶ *Oversampling*: re-amostra dados da classe minoritária a fim de equilibrar os dados de treinamento (SMOTE é um exemplo)
 - ▶ *Undersampling*: decrementa a quantidade de dados da classe majoritária no treinamento
 - ▶ Mover o limiar de separação: altera o parâmetro que indica a classe predita
 - ▶ Técnicas de comitê (*ensemble*)

Avaliação de Modelos: scikit-learn

Avaliação de Modelos

- ▶ A biblioteca scikit-learn disponibiliza diversos mecanismos para avaliação de modelos



<http://scikit-learn.org/stable/modules/classes.html#module->

- ▶ Acurácia (`accuracy_score`)
- ▶ Matriz de Confusão (`confusion_matrix`)
- ▶ F_β score (`fbeta_score`)
- ▶ Curva ROC (`roc_curve`)
- ▶ Área Sob a Curva ROC (`roc_auc_score`)

Avaliação de Modelos

- ▶ Há também implementações de métodos auxiliares para a avaliação
 - ▶ <http://scikit-learn.org/stable/modules/classes.html#module-sklearn>
 - ▶ Separação dos dados/*holdout* (`train_test_split` e `ShuffleSplit`)
 - ▶ Validação Cruzada (`GroupKFold`, *`cross_validate`* e *`cross_val_score`*)
 - ▶ Validação Cruzada Estratificada (`StratifiedKFold`)
- ▶ Métodos para determinação de parâmetros dos modelos
 - ▶ Busca em Grade (`GridSearchCV`)
 - ▶ Busca Aleatória (`RandomizedSearchCV`)
- ▶ Pode-se implementar uma estratégia
- ▶ Análise estatística
 - ▶ Teste-t (`scipy.stats.ttest_ind` e `scipy.stats.ttest_rel`)
 - ▶ Teste de Wilcoxon (`scipy.stats.wilcoxon`)

Avaliação de Modelos

- ▶ Há também implementações de métodos auxiliares para a avaliação
 - ▶ <http://scikit-learn.org/stable/modules/classes.html#module-sklearn>
 - ▶ Separação dos dados/*holdout* (`train_test_split` e `ShuffleSplit`)
 - ▶ Validação Cruzada (`GroupKFold`, `cross_validate` e `cross_val_score`)
 - ▶ Validação Cruzada Estratificada (`StratifiedKFold`)
- ▶ Métodos para determinação de parâmetros dos modelos
 - ▶ Busca em Grade (`GridSearchCV`)
 - ▶ Busca Aleatória (`RandomizedSearchCV`)
- ▶ Pode-se implementar uma estratégia
- ▶ Análise estatística
 - ▶ Teste-t (`scipy.stats.ttest_ind` e `scipy.stats.ttest_rel`)
 - ▶ Teste de Wilcoxon (`scipy.stats.wilcoxon`)

Avaliação de Modelos

- ▶ Há também implementações de métodos auxiliares para a avaliação
 - ▶ <http://scikit-learn.org/stable/modules/classes.html#module-sklearn>
 - ▶ Separação dos dados/*holdout* (`train_test_split` e `ShuffleSplit`)
 - ▶ Validação Cruzada (`GroupKFold`, *`cross_validate`* e *`cross_val_score`*)
 - ▶ Validação Cruzada Estratificada (`StratifiedKFold`)
- ▶ Métodos para determinação de parâmetros dos modelos
 - ▶ Busca em Grade (`GridSearchCV`)
 - ▶ Busca Aleatória (`RandomizedSearchCV`)
- ▶ Pode-se implementar uma estratégia
- ▶ Análise estatística
 - ▶ Teste-t (`scipy.stats.ttest_ind` e `scipy.stats.ttest_rel`)
 - ▶ Teste de Wilcoxon (`scipy.stats.wilcoxon`)