Avaliando o que foi Aprendido

Sumário

- Treinamento, teste, validação
- Predição da performance: Limites de confiança
- Holdout, cross-validation, bootstrap
- Comparando algoritmos: o teste-t
- Predecindo probabilidades:função de perdas
- Medidas de Custos
- O principio de descrição minima

Avaliação: A chave do sucesso

- Confiança no modelo aprendido?
- Error nos dados de treinamento não é um bom indicador do erro em dados futuros.
- Exemplo: 1-NN seria um classificador ótimo!
- Uma solução simples que poderia ser usada se tivéssemos disponível muitos dados:
  - Separe os dados em treinamento e teste
- Porém: Dados rotulados são usualmente limitados
  - Técnicas mais sofisticadas devem ser usadas

Avaliação

- Confiança Estatística das diferenças estimadas em performance ($\rightarrow$ teste de significância)
- Escolhas das medidas de performance:
  - Numero de exemplos corretamente classificados
  - Acurâcia das probabilidades estimadas
- Custos de diferentes tipos de erros

Treinamento e Teste

- Medida de Performance Natural para problemas de classificação: *Taxa de erro*
  - *Sucesso*: a classe da instância é predita corretamente
  - *Erro*: a classe da instância é predita incorretamente
  - Taxa de Erro: proporção de erros feitos em todas as instâncias
- *Substituição do Erro: a taxa de erro é obtida do conjunto de treinamento*
- Muito otimista!

Teste

- *O conjunto de teste: instâncias que não foram usadas no treinamento*
- Suposição: ambos treinamento e teste são exemplos representativos do problema estudado
- Os conjuntos podem ser diferentes naturalmente
- Exemplo: classificadores construídos usando consumidores de dois diferentes cidades A e B
- Para estimar a performance do classificador aprendido de A testar em B, os consumidores podem ser completamente diferentes

Observação sobre Ajuste de Parametros

- É importante que o conjunto de teste não seja usado de nenhuma forma para criar o classificador
- Alguns algoritmos operam em dois estágios:
- Estagio 1: construir a estrutura básica
- Estagio 2: Otimize os parâmetros
- O conjunto de teste não pode ser usado para ajuste
- Procedimento usar 3 conjuntos: *dados de treinamento*, *dados de validação*, e teste
- Os dados de validação são usados para ajustar os parametros

Classificador final

- Uma vez que a avaliação esta completa, todos os dados podem ser usados para construir o classificador final
- Geralmente, quanto maior o conjunto de treinamento melhor o classificador
- Quanto maior o conjunto de teste mais precisa é a estimação do erro
- *Procedimento Holdout* : método de dividir os dados originais em treinamento e teste.
- Dilema: idealmente ambos conjuntos devem ser grandes!

Predizendo Performance

- Assuma que o erro estimado é 25%. Confiança do erro real?
  - Depende do numero de dados do conjunto de teste
- Predição é como jogar uma moeda (com bias)
  - “Cara” é um “sucesso”, “coroa” é um “erro”
- Em estática, uma sucessão de eventos independentes como estes são chamados de processos de Bernoulli
  - A teoria estatística -> intervalos de confiança

Intervalos de Confiança

- *p* esta dentro de um intervalo especifico com uma confiança
- Exemplo: *S*=750 sucessos em *N*=1000
- Taxa de sucesso estimada: 75%
- Taxa de acerto real?
- Resposta: com 80% de confiança *p em* [73.2,76.7]
- Outro exemplo: *S*=75 em *N*=100
- Taxa de sucesso estimada: 75%
- Com 80% de confiança *p em* [69.1,80.1]

Media e Variança

- Media e Variância, Bernoulli trial:
p, p (1–p)
- Taxa de sucesso esperada $f=S/N$
- Media e Variância para f : p, p (1–p)/N
- Para grandes N, f segue uma distribuição Normal
- c% Intervalo de confiança [$-z \leq X \leq z$] para uma variável aleatória com media 0 é dado por:

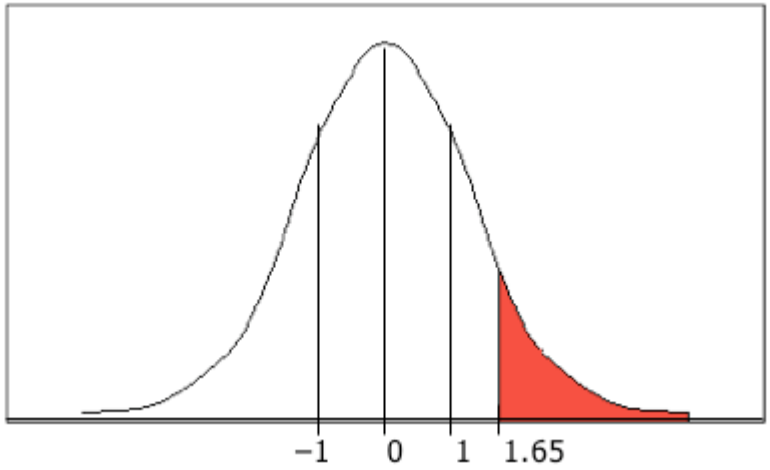
$$Pr[-z \leq X \leq z] = c$$

- Com uma distribuição simétrica:

$$Pr[-z \leq X \leq z] = 1 - 2 \times Pr[x \geq z]$$

Limites de Confiança

- Limites de Confiança para uma variável com distribuição normal com media 0 e variância 1:



Pr[$X \geq z$]	z
0.1%	3.09
0.5%	2.58
1%	2.33
5%	1.65
10%	1.28
20%	0.84
40%	0.25

- Assim:

$$Pr[-1.65 \leq X \leq 1.65] = 90\%$$

Para usar a tabela devemos ter f com media 0 e variância 1

Transformando *f*

- Valores transformados para f:

$$\frac{f-p}{\sqrt{p(1-p)/N}}$$

(isto é subtrair a media e dividir pelo desvio padrão)

- Resultando:

$$Pr\left[-z \leq \frac{f-p}{\sqrt{p(1-p)/N}} \leq z\right] = c$$

- Solucionando p:

$$p = \left(f + \frac{z^2}{2N} \mp z\sqrt{\frac{f}{N} - \frac{f^2}{N} + \frac{z^2}{4N^2}}\right) / \left(1 + \frac{z^2}{N}\right)$$

Exemplos

- f = 75%, N = 1000, c = 80% (z = 1.28):

$$p \in [0.732, 0.767]$$

- f = 75%, N = 100, c = 80% (z = 1.28):

$$p \in [0.691, 0.801]$$

- Note que a distribuição norma; é valida somente para valores de N grandes (isto é, $N > 100$)
- f = 75%, N = 10, c = 80% (z = 1.28):

$$p \in [0.549, 0.881]$$

Estimação com Holdout

- O que fazer se os dados são limitados?
- O método de *holdout* reserva um conjunto para teste e o demais para treinamento
  - Usualmente: 1/3 teste
- Problema: os exemplos podem não ser representativos
  - Exemplos: a classe pode não existir nos exemplos de teste
- Versões mais avanzadas usam e*stratificação*
  - Assegura que cada classe é representada com a mesma distribuição em ambos conjuntos

Métodos de Houldout repetido

- As estimativas podem ser mais confiáveis se o processo de Holdout é repetido com diferentes amostras
  - A cada iteração, uma certa proporção do conjunto é aleatoriamente selecionada para treinamento (possivelmente com estratificação)
  - As taxas de erros dos diferentes conjuntos de teste e é feita a media
- *Isto é chamado o método de holdout*
- Ainda não é ótimo: os conjuntos de teste podem ser sobrepostos
  - Como prevenir?

Validação Cruzada

- Validação cruzada previne o overlap
  - 1 passo: separe os dados em *k subconjuntos de igual tamanho*
  - 2 passo: use cada um dos subconjuntos uma vez para teste e os demais para treinamento
- *k-fold cross-validation*
- Primeiro os subconjuntos podem ser estratificados
- E feita a media com os diferentes erros

Cross-validation

- Método de Avaliação Standard: estratificado 10-fold cross-validation

Porque 10?

  - Experimentos extensivos tem mostram que esta é a melhor forma de obter uma estimativa precisa
  - Existem também algumas evidencias teóricas
- A estratificação reduz a variância estimada
- Melhor é: cross-validation estratificada repetida
  - Isto é 10-fold cross-validation é repetida 10 vezes e a media dos resultados é usada.

Leave-One-Out cross-validation

- Leave-One-Out:
  uma forma particular de cross-validation:
  - O numero de folds é o numero de instâncias
  - Isto é, para *n* instâncias de treinamento, construa n classificadores
  - Faz o melhor uso dos dados
- Não envolve amostragem aleatória
- Muito caro computacionalmente
  - (exceção: KNN)

Leave-One-Out-CV e Estratificação

- Desvantagens de Leave-One-Out-CV: A estratificação não é possível
- Garante uma amostra não-estratificada porque só tem uma instância no teste!
- Exemplo extremo: um conjunto aleatório separado igualmente em 2 classes
  - Melhor indutor prediz a classe majoritária
  - 50% de acuracia em dados novos
  - Leave-One-Out-CV estimativa é 100% de erro!

Bootstrap

- CV usa amostragem sem reposição
  - A mesma instância, uma vez selecionada, não pode ser selecionada de novo para um conjunto de treinamento ou teste particular
- B*ootstrap* usa amostragem com reposição para formar o conjunto de treinamento
  - Amostre um conjunto com *n* instâncias *n* vezes com reposição para formar um novo conjunto
  - Use estes dados como treinamento
  - Use as instâncias do conjunto original que não estão treinamento com teste.

0.632 bootstrap

- *0.632 bootstrap*
  - Uma instância tem uma probabilidade 1–1/*n* de não ser escolhida
  - O conjunto de treinamento conterá aproximadamente 63.2% das instâncias

Erro Estimado com Bootstrap

- O erro estimado no conjunto de dados de teste será pessimista
- Treinamento em somente ~63% das instâncias
- Portanto, combinado com o erro de reposição
- O erro de reposição tem menos pesos que o erro no conjunto de teste
- Repita o processo varias vezes com diferentes amostras com reposição e faça a media dos resultados

Comentários bootstrap

- Provavelmente a melhor maneira de ter uma estimativa para pequenos conjuntos de dados
- Porém tem alguns problemas
  - Considere os dados aleatórios anteriores
  - Um memorizador perfeito terá
    0% de erro de reposição
    ~50% erro em dados de teste
  - Bootstrap estimado para este classificador :
    $err = 0.632 \times 50\% \cdot 0.368 \times 0\% = 31.6\%$
  - True Erro esperado: 50%

Comparando algoritmos

- Perguntas Freqüentes: que algoritmo tem melhor performance melhor ?
- Note: Depende do domínio!
- Compare as estimativas 10-fold CV
- Geralmente suficiente
- Porém, em pesquisas de aprendizado de máquina?
  - Precisa-se provar que o algoritmo é realmente melhor

Comparando Algoritmos II

- A é melhor que B num domínio particular
  - Para uma quantidade de dados de treinamento
  - Em media, para todos os possíveis conjuntos de treinamento
- Assuma que temos uma quantidade infinita de dados de um domínio:
  - Amostre infinitamente muitos conjuntos de tamanhos específicos.
  - Obtenha estimadas de cross-validation em cada dos conjuntos para cada algoritmo
  - Verifique se a acuracia do algoritmo A é melhor que a de B

Teste t

- Na pratica os dados e as estimativas são limitados.
- *O teste de Student' t* é usado para determinar quando duas amostras são significativamente diferentes
- Em nosso caso as amostras são de estimativas cross-validation para diferentes conjuntos de diferentes domínios
- Use um teste t pareado porque as amostras individuas são pareadas
  - O mesmo CV é aplicado duas vezes

Distribuição das Medias

- $x_1\ x_2\ \ldots\ x_k$ e $y_1\ y_2\ \ldots\ y_k$ *são* $2k$ amostras para k diferentes conjuntos
- m_x e m_y são as medias
- Com suficientes amostras, as medias são um conjunto independentes normalmente distribuídas
- As variâncias das medias são σ_x^2/k e σ_y^2/k

- Se μ_x *e* μ_y são as medias reais então estão aproximadamente normalmente distribuídas com media 0 e variância 1
  1

Distribuição de Student

- Com pequenas amostras(k < 100) a media segue a distribuição de Student com k–1 *graus de liberdade*
- Limites de confiança:

Supondo 10 estimações

9 graus de liberdade

Pr[$X \geq z$]	z
0.1%	4.30
0.5%	3.25
1%	2.82
5%	1.83
10%	1.38
20%	0.88

distribuição normal

Pr[$X \geq z$]	z
0.1%	3.09
0.5%	2.58
1%	2.33
5%	1.65
10%	1.28
20%	0.84

Distribuição das Diferenças

- Seja $m_d = m_x - m_y$
- A diferença das medias(m_d) também tem uma distribuição de Student' com k–1 graus de liberdade
- Seja σ_d^2 as variância das diferenças
- A versão padrão de m_d é chamada de estatística t:

$$t = \frac{m_d}{\sqrt{\sigma_d^2 / k}}$$

- Usamos t para realizar o t-test

Realizando o Teste

- Fixe o nível de significância
  - Se a diferença é significante no nível α%, isto é (100-α)% de chances que a media real sejam diferentes
- Divida o nível de significância por 2 porque o teste é de duas caudas
  - Isto é a diferença real pode ser +ve ou – ve
- Veja o valor de z que corresponde a $\alpha/2$
- Se $t \leq -z$ ou $t \geq z$ então a diferença é significante
  - Isto é a hipótese nula (que não existe diferença) pode ser rejeitada

Custo

- A matriz de *confusão*:

		Predicted class	
		Yes	No
Actual class	Yes	True positive	False negative
	No	False positive	True negative

Estatística kappa

- Predição do alg. vs. Predição aleatória
-

		Predicted class			
		a	*b*	*c*	*total*
Actual class	*a*	88	10	2	*100*
	b	14	40	6	*60*
	c	18	10	12	*40*
	total	*120*	*60*	*20*	

		Predicted class			
		a	*b*	*c*	*total*
Actual class	*a*	60	30	10	*100*
	b	36	18	6	*60*
	c	24	12	4	*40*
	total	*120*	*60*	*20*	

- Numero de sucessos: soma da diagonal(*D*)
- *Estatística Kappa*:
- $$\frac{D_{observed} - D_{random}}{D_{perfect} - D_{random}}$$
- mede a melhoria sobre um alg. aleatório

Classificação com custos

- Matrizes de custos:

		Predicted class	
		yes	*no*
Actual class	*yes*	0	1
	no	1	0

		Predicted class		
		a	*b*	*c*
Actual class	*a*	0	1	1
	b	1	0	1
	c	1	1	0

- A taxa de sucesso é substituída pelos custos das predições
- Os custos são dados pelas entradas nas matrizes