Dados Qualitativos: categorias ou grupos. Subdivididos em nominal e ordinal.

Dados Quantitativos: Representam números e podem ser subdivididos em:

- Discreto: Valores contáveis e finitos (ex: número de pessoas, quantidade de produtos vendidos).

- Contínuo: Valores em um intervalo contínuo (ex: altura, peso, tempo).

Escalas de Medida

- Escala Nominal: Identifica categorias sem ordem (ex: cores, gêneros). Valores com nomes diferentes.

- Escala Ordinal: Ordena categorias com hierarquia (ex: classificações, níveis educacionais).

- Escala Intervalar: Mede diferenças com intervalos iguais, mas sem verdadeiro zero (ex: temperatura em Celsius).

- Escala Racional: Possui verdadeiro zero, permitindo proporções (ex: peso, altura).

Tipos de Algoritmos

- Classificação: Predição de categorias ou classes (ex: identificar se um e-mail é spam ou não).

- Regressão: Predição de valores contínuos (ex: prever o preço de uma casa).

- Clusterização: Agrupamento de dados sem rótulos pré-definidos (ex: segmentação de clientes).

- Redes Neurais: Modelos inspirados no cérebro humano, usados em diversas aplicações como reconhecimento de voz.

- Árvores de Decisão: Modelos baseados em divisões recursivas dos dados para fazer previsões.

Tipos de Aprendizado

- Aprendizado Supervisionado: Usa dados rotulados para treinar o modelo (ex: classificação, regressão).

- Aprendizado Não Supervisionado: Trabalha com dados não rotulados (ex: clusterização).

- Aprendizado por Reforço: O modelo aprende por meio de interação com o ambiente, recebendo recompensas ou penalidades (ex: jogos, controle de robôs).

1. Você consegue pensar em uma situação em que números de identificação seriam úteis para previsão?

Se o número estiver associado quando o dado foi criado, e quando algo pode acontecer com ele.

Exemplo:

ID_Estudante é um bom dado para prever data de graduação.

2. Diferencie entre ruído e outliers. Certifique-se de considerar as seguintes perguntas:
   (a) O ruído é alguma vez interessante ou desejável?
   (b) Objetos de ruído podem ser outliers?
   (c) Objetos de ruído são sempre outliers?
   (d) Outliers são sempre objetos de ruído?
   (e) O ruído pode transformar um valor típico em incomum, ou vice-versa?

   Ruído: São dados aleatórios ou irrelevantes que não têm significado útil e podem atrapalhar a análise. Exemplo: erros de medição.

Outliers: São pontos de dados que se destacam por serem muito diferentes da maioria dos dados. Podem ser úteis, pois às vezes indicam algo interessante, como uma descoberta ou anomalia.

a) Às vezes, o ruído pode ser desejável em simulações para criar cenários realistas. No entanto, normalmente ele é indesejado.

b) Sim, objetos de ruído podem ser outliers se forem muito diferentes do padrão.

c) Não, nem todos os objetos de ruído são outliers. Por exemplo, ruídos pequenos podem estar dentro do padrão.

d) Não, outliers podem ser legítimos e conter informações úteis, como uma alta pontuação em uma prova fora do normal.

e) Sim, ruído pode distorcer os valores típicos, tornando-os incomuns, ou mascarar outliers, fazendo-os parecer normais.

3. As seguintes características são medidas para membros de um rebanho de elefantes asiáticos: peso, altura, comprimento das presas, comprimento da tromba e área das orelhas. Com base nessas medidas, que tipo de medida de similaridade vista em aula você usaria para comparar ou agrupar esses elefantes? Justifique sua resposta e explique quaisquer circunstâncias especiais.

Resposta:
Eu usaria a distância Euclidiana. A distância Euclidiana é adequada porque mede diretamente a diferença entre as características dos elefantes. Porém, temos atributos (peso, altura, etc.) em diferentes escalas. A magnitude importa: características maiores (como peso) não podem dominar os cálculos. Se quisermos destacar relações proporcionais ou eliminar o impacto das unidades de medida, é necessário normalizar ou padronizar os dados.

Calcule a distância de Hamming e a similaridade de Jaccard entre os seguintes vetores binários:
x(1) = 0101010001
x(2) = 0100011000

- Distância de Hamming: Contamos os bits diferentes entre os dois vetores:
  Comparação bit a bit: Diferenças em posições 4, 7, 10 ⟶ Hamming = 3.
- Similaridade de Jaccard:
  Numeros de 1s em comum/ numero total de 1s. 2/7

(b) Qual abordagem, Jaccard ou distância de Hamming, é mais semelhante ao Coeficiente de Similaridade Simples (Simple Matching Coefficient) e qual abordagem é mais semelhante à medida de cosseno? Explique. (Nota: A medida de Hamming é uma distância, enquanto as outras três são similaridades, mas não deixe isso confundir você.)

Hamming ↔ Coeficiente Simples: Mais semelhante ao Coeficiente de Similaridade Simples, pois considera todas as posições (0s e 1s).

Jaccard ↔ Cosseno: Mais semelhante ao Cosseno, pois ambos ignoram 0s e se concentram apenas nos 1s para calcular similaridade.

(c) Suponha que você esteja comparando quão semelhantes dois organismos de espécies diferentes são em termos do número de genes que compartilham. Descreva qual medida, Hamming ou Jaccard, seria mais apropriada para comparar a composição genética dos dois organismos. Explique. (Assuma que cada animal é representado como um vetor binário, onde cada atributo é 1 se um determinado gene estiver presente no organismo e 0 caso contrário.)

Usaria Jaccard. Foca apenas nos genes compartilhados (1s em comum), ignorando os genes ausentes (0s). Faz sentido para organismos diferentes, pois queremos medir o que eles têm em comum, sem considerar o que falta.

(d) Se você quisesse comparar a composição genética de dois organismos da mesma espécie, por exemplo, dois seres humanos, usaria a distância de Hamming, o coeficiente de Jaccard ou uma medida diferente de similaridade ou distância? Explique. (Nota: dois seres humanos compartilham > 99,9% dos mesmos genes.)

Use Hamming ou uma variação como distância percentual. Como humanos compartilham > 99,9% dos genes, queremos medir diferenças precisas. Hamming destaca essas pequenas diferenças específicas em vez de focar apenas nos 1s compartilhados, que já são quase todos.

6) Para os seguintes vetores, x e y, calcule as similaridades ou distâncias indicadas.
(a) x = (1, 1, 1, 1), y = (2, 2, 2, 2) – cosseno, correlação, Euclidiana
(b) x = (0, 1, 0, 1), y = (1, 0, 1, 0) – cosseno, correlação, Euclidiana, Jaccard

r=1: Correlação perfeita e positiva. À medida que XXX aumenta, YYY também aumenta de forma linear.

r=−1: Correlação perfeita e negativa. À medida que XXX aumenta, YYY diminui de forma linear.

r=0r = 0r=0: Nenhuma correlação linear. As variáveis não possuem relação linear significativa.

0<r<10 < r < 10<r<1: Correlação positiva parcial.

−1<r<0-1 < r < 0−1<r<0: Correlação negativa parcial.

7)
(a) Qual é o intervalo de valores possível para a medida de cosseno?

O intervalo de valores para a medida de cosseno é de 0 a 1, se os vetores forem não-negativos, ou de -1 a 1, se vetores com valores negativos forem permitidos.
(b) Se dois objetos têm uma medida de cosseno igual a 1, eles são idênticos? Explique.

Não, dois objetos com medida de cosseno igual a 1 não precisam ser idênticos. Isso significa que eles têm a mesma direção no espaço vetorial, mas podem ter magnitudes diferentes.
(c) Qual é a relação entre a medida de cosseno e a correlação, se houver alguma? (Dica: observe medidas estatísticas como média e desvio padrão nos casos em que cosseno e correlação são iguais ou diferentes.)

A medida de cosseno e a correlação estão relacionadas, mas não são iguais. A correlação considera a média e o desvio padrão dos dados, enquanto a medida de cosseno apenas considera o ângulo entre os vetores. Quando os vetores são normalizados (ou centralizados e escalados), a medida de cosseno e a correlação podem ser iguais. Caso contrário, elas diferem.

Table 1: Dataset for exercise (8).

$\mathbf{x}_1$	0.5	3.0	4.5	4.6	4.9	5.2	5.3	5.5	7.0	9.5
$f(x)$	−	−	+	+	+	−	−	+	−	−

a) Classify the data point x(test) = 5.0 according to its 1-, 3-, 5-, and 9-nearest neighbors using the Euclidean distance and majority vote scheme.

(b) Repeat the previous analysis using the distance-weighted voting approach that weighs each object i with weight $w_i = 1/(dist(x^{(i)}, x^{(test)}))^2$

10) Suponha que você precise classificar pessoas como "atleta" ou "não-atleta", em um cenário onde os atributos preditivos incluem peso (em quilos) e altura (em metros). Explique por que usar o algoritmo k-NN neste cenário sem nenhum pré-processamento não seria uma boa ideia. O que você faria em termos de pré-processamento para tornar o uso do k-NN uma opção viável?

Peso e altura são atributos dentro de intervalos muito diferentes (por exemplo, [0,200] e [0.5,2.5]) e com variâncias distintas. Nesse cenário, as diferenças no peso afetariam o cálculo da distância de forma muito mais drástica do que as diferenças na altura (que, no fim, seriam praticamente desconsideradas). Se a intenção for levar em conta ambos os atributos como igualmente importantes para o aprendizado, uma possível solução seria reescalá-los para o mesmo intervalo (digamos, [0,1]). Não se esqueça de que o pré-processamento por reescalonamento linear é severamente afetado por outliers, então seria necessário primeiro analisar se o conjunto de dados está livre de outliers.

A explicação é que o algoritmo k-NN (k-vizinhos mais próximos) calcula a distância entre os pontos de dados com base nas características fornecidas, como peso e altura, para determinar a classificação. No entanto, as unidades de medida de peso (quilos) e altura (metros) são diferentes em escala. O peso pode variar de 30 a 200 quilos, enquanto a altura varia de 1,5 a 2 metros. Essas diferenças de escala podem fazer com que o atributo de peso domine o cálculo da distância, distorcendo os resultados da classificação.

Para tornar o uso do k-NN uma opção viável, seria necessário realizar o pré-processamento das variáveis. A normalização ou padronização dessas características seria uma solução. A normalização transforma os valores de cada atributo para uma escala comum, geralmente entre 0 e 1, enquanto a padronização (z-score) ajusta os atributos para terem média zero e desvio padrão 1. Dessa forma, ambos os atributos, peso e altura, terão a mesma influência na determinação da distância entre os pontos, permitindo que o k-NN funcione corretamente.

 Por que utilizaríamos um algoritmo iterativo para o problema de regressão linear?

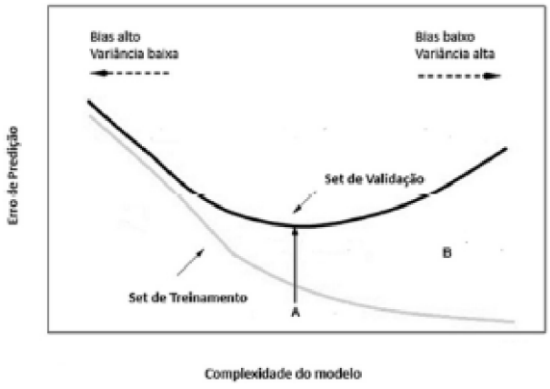
- Um algoritmo iterativo, como o gradiente descendente, é usado na regressão linear quando o número de variáveis ou dados é muito grande, tornando métodos analíticos, como a solução direta da equação normal, ineficientes ou inviáveis devido ao custo computacional. O gradiente descendente ajusta os parâmetros da regressão iterativamente, procurando um mínimo global da função de erro (como o erro quadrático médio). Esse processo de iteração é útil para lidar com grandes volumes de dados e para garantir uma convergência eficiente aos valores ótimos dos parâmetros.

 O que pode acontecer se, ao utilizarmos o algoritmo de otimização via gradiente descendente, escolhermos uma taxa de aprendizado **α** muito pequena? E muito grande?

- Taxa de aprendizado muito pequena: Se a taxa de aprendizado (α) for muito pequena, o algoritmo de gradiente descendente fará atualizações muito lentas nos parâmetros, resultando em um processo de otimização que pode levar muito tempo para convergir para o mínimo global, ou até mesmo ficar preso em um mínimo local sem conseguir melhorar significativamente.
- Taxa de aprendizado muito grande: Por outro lado, se a taxa de aprendizado for muito grande, as atualizações podem ser tão grandes que o algoritmo pode "pular" o mínimo, sem convergir de maneira eficiente. Isso pode levar a oscilações no processo de otimização, em vez de uma convergência estável.

 Como o algoritmo de otimização via gradiente descendente atualiza o vetor de parâmetros Θ?

- O gradiente descendente atualiza os parâmetros (Θ) seguindo a fórmula: $\Theta := \Theta - \alpha \cdot \nabla_\Theta J(\Theta)$ Onde:
    - Θ é o vetor de parâmetros a ser otimizado.
    - α é a taxa de aprendizado.
    - $\nabla_\Theta J(\Theta)$ é o gradiente da função de custo $J(\Theta)$ em relação aos parâmetros Θ, que indica a direção da maior taxa de aumento da função de custo. O algoritmo ajusta os parâmetros na direção oposta ao gradiente para minimizar a função de custo, dando um passo em direção ao mínimo global.



O gráfico mostra a relação entre o erro de predição e a complexidade do modelo, indicando áreas de overfitting (sobreajuste) e underfitting (subajuste), além de destacar a variância e o viés em diferentes pontos.

Agora, vamos analisar as afirmações:

1. "Considerando que a variância é um erro de sensibilidade para pequenas flutuações no conjunto de treinamento, infere-se que um baixo nível de variância pode fazer com que o algoritmo associado a um modelo de aprendizado de máquina perca as relações relevantes

entre os atributos de entrada e a variável de saída, caracterizando o erro de overfitting, percebido na região à direita do ponto A."

    - Errado. A descrição está incorreta. Quando a variância é baixa, o modelo é mais simples e tende a ser mais propenso ao underfitting, não ao overfitting. O overfitting ocorre quando a variância é alta e o modelo se ajusta excessivamente ao conjunto de treinamento, perdendo a capacidade de generalizar para dados não vistos.

2. "Quando se verifica um alto erro no treinamento com valor próximo ao erro na validação, percebido na região à esquerda do ponto A, tem-se um clássico problema de underfitting, caracterizado pelo alto valor do bias."

    - Certo. Isso está correto. Quando o erro no conjunto de treinamento e no conjunto de validação são altos e próximos, o modelo está sofrendo de underfitting, com alto viés (bias), o que significa que o modelo não está conseguindo capturar as complexidades dos dados.

3. "O Set de Treinamento é usado para qualificar o desempenho do modelo, enquanto o Set de Validação é utilizado para criar o modelo de aprendizado de máquina."

    - Errado. Na verdade, o conjunto de treinamento é utilizado para treinar o modelo, ajustando seus parâmetros, enquanto o conjunto de validação é usado para avaliar o desempenho do modelo durante o processo de treinamento, ajudando a ajustar hiperparâmetros, mas não para criar o modelo em si.

4. "A região do gráfico entre as duas curvas, indicada pela letra B, mostra a região de erro de generalização para o modelo de aprendizado de máquina."

    - Certo. A região entre as duas curvas no gráfico, entre o conjunto de treinamento e o conjunto de validação, representa o erro de generalização, que reflete a capacidade do modelo de se generalizar para dados não vistos.

5. "O ensemble denominado bagging tem como foco principal a redução do viés e não da variância, treinando-se os modelos em sequência, tal que os erros dos primeiros modelos treinados são utilizados para o ajuste nos pesos matemáticos dos próximos modelos."

    - Errado. O bagging (Bootstrap Aggregating) é projetado para reduzir a variância, não o viés. Ele treina múltiplos modelos em paralelo usando diferentes subconjuntos do conjunto de dados, e combina suas previsões. Já o boosting é o que foca na redução do viés, treinando modelos em sequência e ajustando os erros dos modelos anteriores.

6. "O algoritmo de backpropagation consiste das fases de propagação e de retro propagação: na primeira, as entradas são passadas através da rede e as previsões de saída são obtidas; na segunda, se calcula o termo de correção dos pesos e, por conseguinte, a atualização dos pesos."

    - Certo. O algoritmo de backpropagation é dividido em duas fases: na propagação direta, as entradas são passadas pela rede para gerar as saídas previstas, e na retropropagação, calcula-se o erro e ajustam-se os pesos com base no gradiente do erro.

7. "As funções de ativação são elementos importantes nas redes neurais artificiais; essas funções introduzem componente não linear nas redes neurais, fazendo com que elas possam aprender mais do que relações lineares entre as variáveis dependentes e independentes, tornando-as capazes de modelar também relações não lineares."
    - Certo. As funções de ativação introduzem não linearidade nas redes neurais, permitindo que elas capturem relações mais complexas, além das lineares, entre as variáveis de entrada e saída.
8. "Em razão de seu bom desempenho, o algoritmo SVM (support vector machines) é invariante à escala dimensional dos conjuntos de dados, o que torna dispensável a padronização e o pré-processamento dos dados."
    - Errado. O SVM (Máquinas de Vetores de Suporte) é sensível à escala das características. Portanto, é fundamental fazer a padronização (ou normalização) dos dados antes de aplicar o SVM para obter um bom desempenho, pois a escala dos dados afeta a forma como os vetores de suporte são escolhidos.

Paradigma de Aprendizado Supervisionado

- Guiado por um "professor" externo
- O professor conhece a tarefa
- Representado por um conjunto de pares (x, d)
- O algoritmo gera um modelo para reproduzir o comportamento do professor.
- Os parâmetros do modelo são ajustados por apresentações sucessivas dos pares (x, d) - Fase de Treinamento
- Após o treinamento, o desempenho do sistema deve ser testado com dados não vistos - Fase de Teste

O que são dados?

- Uma abstração de uma entidade do mundo real.
- Informação é dado que foi processado, estruturado ou contextualizado, de forma a ter significado para os humanos.

- Variável, Atributo e Característica são termos usados indistintamente para indicar uma abstração individual.
- Cada entidade é tipicamente descrita por um número de atributos.
- Um conjunto de dados consiste em dados relacionados a uma coleção de entidades. Cada entidade é descrita em termos de uma lista de atributos.

O que é um Conjunto de Dados?

- Matriz n*m onde n = número de linhas (objetos) e m = número de colunas (atributos)
- Escolhemos os atributos que vão compor o conjunto de dados, que são o resultado do nosso problema e o que acreditamos ser relevante para ele.
- Mais atributos significam mais custo. Para coletar e também custos de desempenho.

Dados, Informação, Conhecimento e Sabedoria/Inteligência

- Dados são criados a partir de abstrações ou medições do mundo real.
- Informação é dado que foi processado, estruturado ou contextualizado, de forma a ter significado para os humanos.
- Conhecimento é informação já interpretada e entendida por um ser humano para que ele possa agir conforme necessário.
- Sabedoria/Inteligência é agir de acordo com o conhecimento.

Valores para atributos

- Os valores dos atributos são números ou símbolos atribuídos a um atributo
- Diferença entre atributo e valor: o mesmo atributo pode ser mapeado com valores diferentes, por exemplo, altura em metros ou pés
- Diferentes atributos podem ser mapeados para o mesmo conjunto de valores.
- A forma como medimos um atributo não pode coincidir com suas propriedades.

Tipos de Atributos

- Nominal (qualitativo) - nomes simples diferentes. Contêm apenas informações suficientes para distinguir uma instância de outra. (operações aceitas: igual ou diferente)
- Ordinal (qualitativo) - informações suficientes para distinguir e ordenar instâncias (igual, diferente, > ou <)

- Intervalar (quantitativo) - atributos onde a diferença entre valores faz sentido. Existe uma unidade de medida com referência arbitrária de zero. (operações aceitas: igual, diferente, > ou <, + ou -)
- Racional (quantitativo) - não apenas a diferença entre valores faz sentido, mas também a razão entre valores (zero é absoluto). (operações aceitas: igual, diferente, > ou <, + ou -, * e /)

Preparação de Dados

Tipos de atributos:

- Atributo contínuo: assume uma quantidade incontável de valores
    - números reais (temperatura, peso, distância)
- Atributo discreto: assume um valor contável (finito ou infinito)
    - estações do ano, cores primárias, código postal, etc.
- Atributos binários: 0 ou 1, V ou F, S ou N...
- Atributos binários assimétricos: assume 2 valores, mas apenas um é relevante (indicando que a instância tem uma característica determinada) (exemplo aqui) - mineração de texto é um cenário clássico.

Exploração preliminar de dados

- Facilita a compreensão de suas características.
- Ajuda a selecionar a melhor técnica de pré-processamento ou aprendizado.
- Utiliza: estatísticas descritivas e visualização.

Estatísticas Descritivas

Permite capturar: frequência dos dados, localização ou tendência central, dispersão ou espalhamento, distribuição ou formato (use termos em inglês adequados aqui).

- Frequência: número de vezes que um atributo assume um valor de dado - frequentemente usado para análise de atributos categóricos.

Medidas de Localização

- Dados categóricos:
    - Moda
- Dados numéricos:
    - Média

    - Mediana
    - Percentil
- Média:
    - Sensível a valores extremos.
    - Bom indicador do centro dos valores do conjunto de dados quando distribuídos simetricamente.
- Mediana:
    - Menos sensível a valores extremos.
    - Necessário ordenar os valores.
    - Complexidade maior que linear no pior cenário.
    - Indica um centro melhor quando a distribuição é enviesada e/ou há valores extremos.
- Média ponderada:
    - "Média truncada"
    - Minimiza os problemas com a média, descartando valores das extremidades. - Percentual p de valores é eliminado - Os dados são ordenados - Elimina p/2% dos valores em cada extremidade.

Quartis

- A mediana divide os dados ao meio.
- Outras medidas usam pontos de divisão diferentes.
- Quartis (em quartos):
    - O primeiro quartil (Q1) é a amostra onde 25% dos valores são inferiores a ele. Também conhecido como o 25º percentil.
    - O segundo quartil (Q2) é a mediana. 50º percentil.

Boxplot

Resumo das informações dos quartis a serem apresentadas em um gráfico chamado boxplot. Percentil: O p-ésimo percentil é um valor de x quando p% dos valores observados são menores do que esse valor.

Medidas de dispersão

Medem a dispersão (ou grau de espalhamento) de um conjunto de dados.
Indicam se um atributo está:

- Amplamente distribuído
- Relativamente concentrado em um ponto específico (ex: média)

Medidas comuns: intervalo, variância, desvio padrão.

- Intervalo: calcula a dispersão máxima. Valor máximo menos o valor mínimo. Não é resistente, pois alterar apenas um dado pode torná-lo arbitrariamente grande.
- Variância (σ^2): Medida preferida de dispersão (dispersão) em análise de dados.
  Denominador (N-1): Conhecido como correção de Bessel, que ajusta o cálculo para uma melhor estimativa da verdadeira variância em uma amostra.
  Segundo momento central: A variância é definida como o segundo momento central da distribuição de dados.
- Desvio Padrão (σ): A raiz quadrada da variância, frequentemente usada junto com a variância para interpretar a dispersão dos dados nas unidades originais de medida.

Assimetria (Skewness)
Definição: Mede a simetria de uma distribuição em torno de sua média.
Independência de escala: A divisão por σ^3 (o desvio padrão elevado ao cubo) torna essa medida independente de escala.
Terceiro momento central: A assimetria é o terceiro momento central padronizado da distribuição.

Curtose (Kurtosis)
Definição: Mede a forma (achatamento ou pico) de uma distribuição.
Quarto momento central: A curtose é o quarto momento central padronizado da distribuição.
Referência da distribuição normal: Para uma distribuição normal padrão, a curtose é 3.
Frequentemente, usa-se a curtose excessiva, que é $\beta(x) - 3$, de modo que uma distribuição normal padrão tem uma curtose excessiva de 0.

Histograma

Definição: Um histograma é uma ferramenta gráfica usada para exibir a distribuição de um conjunto de dados, mostrando a frequência dos pontos de dados dentro de intervalos ou "bins" especificados.
Uso: É especialmente útil para visualizar a curtose (forma ou achatamento) e a assimetria (simetria ou assimetria) da distribuição. A forma do histograma pode revelar se os dados possuem:

- Alta curtose (pico acentuado) ou baixa curtose (distribuição mais plana)
- Assimetria positiva (cauda longa à direita) ou assimetria negativa (cauda longa à esquerda).

Dados univariados e multivariados

- Dados univariados: Examina uma única variável ou característica dos dados para tirar conclusões.
- Dados multivariados: Examina múltiplas variáveis para entender as relações e interações complexas entre elas. Adquirimos cada atributo separadamente e depois agregamos.

A dispersão de um conjunto de dados pode ser capturada por uma matriz de covariância. Cada elemento é a covariância de um par de atributos.
A covariância não indica claramente a relação entre pares de atributos. Podemos usar a correlação, que indica uma relação linear entre duas variáveis.
É preferível explorar os dados usando correlação, em vez de covariância.

Correlação (Pearson)

- Varia de -1 a 1.
- A magnitude dos vetores é ignorada quando normalizada pela variação.
- Ignora a média e a variabilidade.
- Uma correlação de 1 ou -1 significa que xj e xk têm uma relação linear perfeita (positiva ou negativa).
- Uma correlação de 0 significa que não há relação linear.

Transformação de Dados

- Conversão de valores simbólicos para numéricos.
- Conversão de valores numéricos para simbólicos.

Conversão de Valores Categóricos
Muitos algoritmos de aprendizado de máquina (ML), como Redes Neurais e Máquinas de Vetores de Suporte (SVM), funcionam apenas com variáveis numéricas. Variáveis categóricas devem ser convertidas para formato numérico. A abordagem de conversão depende de haver ou não uma ordem inerente nas categorias:

- Variáveis Nominais: A conversão geralmente usa codificação one-hot, onde cada categoria é representada por um vetor binário.
- Variáveis Ordinais: A conversão normalmente usa codificação inteira ou codificação de alvo, preservando a ordem para o modelo.

Conversão de Valores Ordinais - Mantendo a Ordem:
Para variáveis ordinais, a ordem dos valores deve ser preservada de alguma forma.

- Atribuir valores inteiros crescentes para representar a ordem (exemplo: {frio, morno, quente} = {1, 2, 3}).

Problemas Potenciais: Esse método pode introduzir distorções nas diferenças relativas entre categorias, pois as diferenças são subjetivas e podem não refletir verdadeiramente as distâncias entre os conceitos.

Conversão de Valores Nominais
Atributos nominais não possuem uma ordem inerente.

- Método de Conversão: Normalmente, é tratado por binarização, usando codificação one-hot (codificação 1 de n) ou codificação binária inteira.

One-Hot Encoding (Codificação 1 de n):
Atribui um atributo binário exclusivo para cada categoria, com um atributo configurado como 1 e todos os outros como 0.
Exemplo: Codificando {amarelo, vermelho, verde, azul, laranja, branco}, resulta em um vetor binário para cada cor.
Desvantagem de Dimensionalidade: Isso pode gerar um grande número de atributos, levando à maldição da dimensionalidade.
Vantagens:

- Equidistância: Mantém distâncias iguais entre quaisquer dois vetores binários.
- Independência: Atributos binários são não correlacionados.
- Assimetria: Atributos binários são assimétricos, o que é necessário para alguns algoritmos de ML.
- Representação de Modo: O modo de um atributo nominal corresponde ao atributo binário com o maior número de 1s.

Codificação Binária Inteira:
Método: Cada valor nominal é primeiramente atribuído a um número inteiro, e depois convertido para representação binária.
Vantagens:

- Exige menos atributos binários ($\log_2(n)$).
  Desvantagens:
- Diferenças Não Uniformes: As diferenças entre os valores não são as mesmas nas representações inteiras ou binárias, o que pode introduzir correlações entre os atributos.
- Compatibilidade com Algoritmos: Pode ter um desempenho ruim com certos algoritmos de ML devido às correlações induzidas e às distâncias desiguais entre as categorias.

. Tarefas de Aprendizado

Tarefas preditivas: algoritmos são aplicados a um conjunto de dados de treinamento rotulado para induzir um modelo preditivo capaz de prever, para um novo objeto representado pelos valores de seus atributos preditivos, o valor de seu atributo alvo.

Presença de um supervisor externo: sabemos qual é o atributo resultante.
Discreto se for Classificação. Contínuo se for Regressão.

Tarefas descritivas: em vez de prever um valor, os algoritmos extraem padrões dos valores preditivos do conjunto de dados. Aprendizado não supervisionado.

- Clustering (procura por objetos similares dentro de um conjunto de dados), Associação (associa valores de um subconjunto de atributos preditivos a valores de outro subconjunto).
- Resumir (procura uma descrição simples e compacta para um conjunto de dados).

Outros tipos de aprendizado:

- Aprendizado semi-supervisionado: utilizado quando os dados não são rotulados, mas algumas restrições sobre os dados são conhecidas. (adicionar exemplo aqui)
- Aprendizado por reforço: reforça ou recompensa uma ação considerada positiva e pune ações negativas. (adicionar exemplo aqui)

2. Aprendizado de Máquina e Indução de Modelos

No Aprendizado de Máquina, programamos dispositivos para aprender com a experiência. Para isso, usamos um princípio de inferência chamado INDUÇÃO (extrair conclusões gerais de um conjunto de exemplos particulares).

Os algoritmos de aprendizado de máquina devem ser capazes de lidar com dados imperfeitos. Muitos conjuntos de dados incluem ruídos, dados inconsistentes, dados faltantes e dados redundantes.
Os algoritmos devem ser robustos a esses problemas. Mas, caso não sejam, existem técnicas de pré-processamento que ajudam a identificar, reduzir e até eliminar esses problemas.

Quando aplicado aos dados, o algoritmo deve ser capaz de prever algo, para que possa ser utilizado em outros objetos do mesmo domínio, mas fora do conjunto de dados.
Isso é necessário para que o modelo seja válido para novos objetos.

Overfitting: Um modelo com baixa capacidade de generalização tem uma regra que está super ajustada aos dados. Ou seja, o modelo memorizou ou se especializou no conjunto de dados de treinamento.

Underfitting: Quando o modelo tem baixa capacidade preditiva para o conjunto de dados de treinamento. Isso acontece quando os dados de treinamento não são suficientemente representativos ou quando o modelo é simples demais e não captura os padrões existentes nos dados.

3. Viés Indutivo

Um algoritmo de aprendizado de máquina busca um modelo capaz de modelar a relação entre os atributos preditivos e um atributo alvo.
Cada algoritmo representa os modelos possíveis que podem ser encontrados usando seu próprio formato ou linguagem. Exemplo: redes neurais artificiais (valores reais associados ao peso das conexões da rede), árvores de decisão (estrutura de árvore, cada nó é uma pergunta referente ao valor de um atributo, e cada nó está associado a uma classe).

A linguagem ou representação utilizada define uma preferência, ou viés, na representação do algoritmo, restringindo os modelos que ele pode encontrar.

- A forma como um algoritmo busca um modelo: viés de busca

Cada algoritmo tem 2 vieses: viés de representação e viés de busca.

- O viés é necessário para restringir os modelos a serem avaliados no espaço de busca. Sem viés, não haveria aprendizado/generalização.

Indução de Árvores de Decisão

- Descobrir a 'árvore ótima' é um problema NP-difícil.
- Muitas heurísticas para gerar árvores:
    - De cima para baixo
    - De baixo para cima
    - Híbrido
    - Algoritmos Evolutivos

Indução de Cima para Baixo

- Algoritmo de Hunt
    - Suponha que D_{t} seja o conjunto de exemplos de treinamento que chegam ao nó t.
    - Suponha que y_{t} são os rótulos das classes.
- Passo 1:
    - Se todas as instâncias em D_{t} pertencem à mesma classe C_{t}, então t é um nó folha rotulado como C_{t}.
- Passo 2:
    - Se D_{t} contém instâncias de mais de uma classe, é selecionado um teste em um atributo específico para particionar os registros em subconjuntos menores. Um nó é

criado para cada resultado do teste, e as instâncias em DtD_tDt são distribuídas entre esses nós com base nos resultados. O algoritmo é aplicado recursivamente para cada nó gerado.

O algoritmo de Hunt constrói árvores de decisão começando da raiz e indo para baixo. Se todos os exemplos em um nó pertencem à mesma classe, esse nó se torna uma folha rotulada. Caso contrário, o algoritmo escolhe um atributo para dividir os dados e continua recursivamente para cada subconjunto.

Estratégia Recursiva: O processo de construção da árvore se repete, quebrando os dados em partes menores.

Estratégia Gananciosa: Os dados são divididos escolhendo o melhor atributo que dá o benefício mais imediato em cada passo.

- Decisões importantes a serem tomadas:
    - Como dividir os dados: Decida qual atributo usar para dividir os dados.
    - Quando parar de dividir: Decida quando parar de dividir e considerar o nó como final.

Como filtrar os dados com base em um atributo?

- Depende do tipo de atributo:
    - Nominal
    - Ordinal
    - Contínuo
- Depende do número de divisões desejadas:
    - Binária
    - Múltipla

Dividindo para atributos categóricos nominais

- Múltipla: Dividir com base no número de categorias
- Binária: Agrupar as categorias em dois subconjuntos. É necessário encontrar a divisão ideal.

Dividindo para atributos contínuos

- Múltipla: Discretizar os valores em intervalos
- Binária: Definir um ponto de divisão

Critérios de Parada para Indução de Cima para Baixo

- Pare de expandir nós quando:

    - Todas as instâncias pertencem à mesma classe (homogeneidade de classe)
    - Todos os valores dos atributos são idênticos (homogeneidade das instâncias)
    - Um valor satisfatório para o critério de divisão for alcançado (parâmetro)
    - A profundidade máxima for atingida (parâmetro)

Perguntas

- As árvores de decisão não têm um viés de restrição (ou seja, são capazes de representar qualquer função de classificação de dados). Qual é o limite inferior da taxa de erro que as árvores construídas usando o critério de homogeneidade de classe podem alcançar nos dados de treinamento?
- Isso significa que as árvores de decisão são mais propensas a underfitting ou overfitting?

Vantagens e Desvantagens das Árvores de Decisão

- Vantagens:
    - Fácil de entender (amplamente usada por médicos!)
    - Pode gerar regras com base nas árvores
    - Custo baixo para gerar o modelo
    - Extremamente rápida para classificar novas instâncias
- Desvantagens:
    - Pode se tornar muito grande
    - Propensa a overfitting (ajustar demais aos dados)
    - Gera apenas hiperplanos paralelos aos eixos
    - Portanto, não lida bem com atributos correlacionados (por quê?)
    - A solução localmente ótima pode estar longe da ótima global