Item do edital: Ingestão de dados estruturados

**Parágrafo 1:** A ingestão de dados estruturados envolve o processo de extrair, transformar e carregar dados organizados em formatos específicos, como planilhas, arquivos CSV ou bancos de dados relacionais. Esses dados geralmente seguem um esquema predefinido, com campos e tipos de dados consistentes. A ingestão é essencial para integrar dados de diversas fontes em um sistema centralizado, permitindo análises e insights mais fáceis.

**Parágrafo 2:** O processo de ingestão de dados estruturados geralmente envolve etapas de extração, transformação e carregamento (ETL). A extração recupera dados das fontes, enquanto a transformação limpa, normaliza e organiza os dados para garantir compatibilidade. Finalmente, o carregamento insere os dados transformados em um data warehouse ou outro destino de armazenamento. Uma fórmula comum usada na transformação é a conversão de unidades, como converter libras em quilogramas usando a fórmula: quilogramas = libras \* 0,4536.

**Parágrafo 3:** A ingestão de dados estruturados oferece várias vantagens. Melhora a qualidade dos dados, reduz a redundância e garante a consistência. Além disso, facilita o acesso e a análise dos dados, permitindo que as empresas obtenham insights mais profundos de suas operações. A ingestão de dados estruturados é uma etapa crucial no gerenciamento de dados modernos e é essencial para organizações que buscam tomar decisões informadas com base em dados confiáveis.

Item do edital: Ingestão de dados semiestruturados

**Parágrafo 1:** A ingestão de dados semiestruturados envolve o processamento e importação de dados que não se encaixam em um formato de dados estruturado tradicional (por exemplo, tabelas de banco de dados). Esses dados podem vir de várias fontes, como documentos de texto, arquivos de log, XML ou JSON. A ingestão de dados semiestruturados é essencial para organizações que precisam extrair insights de uma ampla gama de fontes de dados.

**Parágrafo 2:** Um desafio comum na ingestão de dados semiestruturados é a variabilidade dos dados. Os dados podem ter diferentes estruturas, semânticas e formatos. Para lidar com essa variabilidade, várias técnicas são usadas, como normalização de dados, extração de recursos e aprendizado de máquina. A normalização de dados converte os dados em um formato comum, enquanto a extração de recursos seleciona atributos relevantes dos dados. O aprendizado de máquina pode ser usado para identificar padrões e inferir estrutura dos dados semiestruturados.

**Parágrafo 3:** Uma fórmula comumente usada para avaliar a qualidade da ingestão de dados semiestruturados é a **precisão**:

Precisão = (Verdadeiros Positivos) / (Verdadeiros Positivos + Falsos Positivos)

onde:

* Verdadeiros Positivos: Número de dados ingeridos corretamente
* Falsos Positivos: Número de dados ingeridos incorretamente

Item do edital: Ingestão de dados não estruturados

A ingestão de dados não estruturados é o processo de coleta e integração de dados que não estão organizados em um esquema ou formato predefinido. Esses dados podem incluir texto, imagens, vídeos, áudio, dados de sensores e outros tipos de conteúdo não estruturado.

A ingestão de dados não estruturados é um desafio, pois esses dados podem ser difíceis de processar e analisar. No entanto, existem várias ferramentas e técnicas que podem ser usadas para automatizar o processo de ingestão de dados não estruturados. Uma abordagem comum é usar um pipeline de ingestão de dados, que é uma série de etapas que prepara e transforma os dados não estruturados para análise.

A fórmula para ingestão de dados não estruturados é:

\*\*Dados não estruturados \*\* + **Pipeline de ingestão \*\* =** Dados prontos para análise \*\*

Item do edital: Ingestão de dados em lote (batch)

A ingestão de dados em lote é um processo de coleta e inserção de um grande volume de dados em um sistema de destino em um único lote. Isso difere da inserção de dados em tempo real, na qual os dados são inseridos individualmente assim que são gerados. A ingestão de dados em lote é frequentemente usada para processar dados históricos ou dados que foram coletados ao longo de um período de tempo.

Para realizar a ingestão de dados em lote, os dados são primeiro coletados de fontes de dados, como arquivos CSV, bases de dados ou APIs. Esses dados são então organizados e formatados em um formato compatível com o sistema de destino. Por fim, os dados são inseridos no sistema de destino usando uma ferramenta ou script de ingestão de dados.

O desempenho da ingestão de dados em lote pode ser medido usando várias métricas, como throughput (taxa de transferência de dados por unidade de tempo), latência (tempo gasto para ingerir os dados) e confiabilidade (porcentagem de dados inseridos com sucesso). A eficiência da ingestão de dados em lote também pode ser melhorada por meio do uso de técnicas como particionamento, compactação e paralelismo.

Item do edital: Ingestão de dados em streaming

**Parágrafo 1:**

A ingestão de dados em streaming envolve a captura e o processamento contínuo de grandes quantidades de dados gerados em tempo real. Os dados de streaming podem ser provenientes de várias fontes, como dispositivos de IoT, mídias sociais ou transações financeiras. Ao contrário dos dados em lote, que são processados em blocos, os dados de streaming são processados individualmente ou em pequenos lotes à medida que chegam.

**Parágrafo 2:**

Os sistemas de ingestão de dados em streaming usam arquiteturas distribuídas para lidar com o alto volume e a velocidade dos dados. Eles geralmente empregam uma abordagem baseada em pipeline que divide o processo de ingestão em etapas menores e gerenciáveis. Cada etapa pode ser executada em nós diferentes, permitindo escalabilidade horizontal e resiliência. As fórmulas comuns usadas para calcular a taxa de ingestão incluem:

* Taxa de Ingestão = (Número de Eventos / Tempo)
* Latência de Ingestão = Tempo desde o Evento até o Armazenamento

**Parágrafo 3:**

A ingestão de dados em streaming é fundamental para aplicativos em tempo real e análises em larga escala. Ele permite que as organizações respondam rapidamente a eventos, monitorem sistemas e tomem decisões informadas. No entanto, também apresenta desafios, como lidar com dados de vários formatos, garantir a consistência e depurar problemas em sistemas distribuídos. As melhores práticas incluem usar uma plataforma de ingestão especializada, otimizar pipelines de dados e implementar monitoramento e alerta robustos.

Item do edital: Armazenamento de big data

**Parágrafo 1:**

O armazenamento de big data é a prática de gerir e armazenar conjuntos de dados volumosos que ultrapassam as capacidades tradicionais de armazenamento de dados. Esses conjuntos de dados podem incluir informações de várias fontes, como transações financeiras, dados de redes sociais, dados sensoriais de dispositivos IoT e muito mais. O armazenamento de big data requer soluções especializadas e escaláveis que possam lidar com o volume, variedade e velocidade crescentes dos dados.

**Parágrafo 2:**

As soluções de armazenamento de big data empregam uma variedade de tecnologias, incluindo sistemas de arquivos distribuídos (por exemplo, HDFS), bancos de dados NoSQL (por exemplo, Cassandra, MongoDB) e sistemas de gerenciamento de dados em nuvem (por exemplo, Amazon S3, Google Cloud Storage). Cada tecnologia oferece recursos específicos, como escalabilidade horizontal, latência reduzida ou gerenciamento automatizado de dados. Para otimizar o armazenamento de big data, é essencial considerar fatores como o tamanho dos dados, a taxa de crescimento, os padrões de acesso aos dados e a necessidade de processamento em tempo real.

**Parágrafo 3:**

Fórmulas comuns usadas na análise de big data incluem:

* **Volume de dados:** Medido em bytes, terabytes (TB), petabytes (PB) ou exabytes (EB).
* **Variedade de dados:** Diversidade de tipos de dados, como texto, números, imagens e dados JSON.
* **Velocidade de dados:** Medida da rapidez com que os dados são gerados e processados, geralmente expressa como taxa de transferência ou latência.
* **Valor de dados:** Medida da relevância e utilidade dos dados para uma organização específica.

Item do edital: Conceitos de processamento massivo e paralelo

**Parágrafo 1:** O processamento massivo envolve o gerenciamento e análise de grandes conjuntos de dados, compostos por bilhões ou até trilhões de registros ou pontos de dados. Isso requer técnicas especializadas, como processamento distribuído e bancos de dados NoSQL, que fragmentam e distribuem dados em vários servidores para lidar com cargas de trabalho massivas.

**Parágrafo 2:** O processamento paralelo é uma técnica que divide uma tarefa complexa em subtarefas menores e as executa simultaneamente em vários processadores. Isso acelera significativamente os tempos de processamento, especialmente para tarefas que podem ser paralelizadas. A Lei de Amdahl quantifica o potencial de velocidade: S = 1 / ((1-P) + P/N), onde S é a velocidade, P é a porcentagem da tarefa que pode ser paralelizada e N é o número de processadores.

**Parágrafo 3:** O processamento massivo e paralelo frequentemente se sobrepõem, pois os conjuntos de dados massivos geralmente exigem processamento paralelo para gerenciamento e análise eficientes. O processamento de fluxo é uma forma especializada de processamento massivo e paralelo que lida com dados em tempo real, fragmentando o fluxo de dados em pequenos pacotes e processando-os à medida que chegam.

Item do edital: Processamento distribuído

**Parágrafo 1:**

O processamento distribuído envolve a divisão de uma tarefa em subtarefas e sua execução em vários computadores ou nós conectados por meio de uma rede. Esses nós trabalham juntos para resolver o problema, compartilhando dados e coordenando suas atividades. Existem dois tipos principais de processamento distribuído: processamento paralelo e processamento distribuído geograficamente.

**Parágrafo 2:**

No processamento paralelo, os nós são tipicamente conectados em um único local, como um cluster de computadores. Eles trabalham simultaneamente em diferentes partes da subtarefa, melhorando a eficiência e a velocidade de processamento. A eficiência do processamento paralelo pode ser quantificada pela Lei de Amdahl, que é dada por E = 1 - (1 - p) \* (T / (pT + (1-p)\*T)), onde E é a eficiência do paralelismo, p é a proporção da tarefa que pode ser paralelizada, T é o tempo sequencial e T é o tempo paralelo.

**Parágrafo 3:**

No processamento distribuído geograficamente, os nós são distribuídos por uma ampla área geográfica, geralmente conectados pela Internet. Este tipo de processamento é útil para tarefas que requerem acesso a recursos e dados específicos que podem não estar disponíveis em um único local. Os desafios do processamento distribuído geograficamente incluem latência de rede, disponibilidade de recursos e segurança.

Item do edital: Soluções de big data: Arquitetura do ecossistema Spark

**Parágrafo 1:**

O ecossistema Spark se baseia em uma arquitetura de componentes interconectados para processar e analisar quantidades massivas de dados. O mecanismo principal, Apache Spark SQL, fornece uma interface semelhante ao SQL para manipular dados estruturados. Além disso, o Spark Streaming permite o processamento de dados em tempo real, enquanto o Spark MLlib oferece recursos de aprendizado de máquina.

**Parágrafo 2:**

O ecossistema Spark é projetado para computação distribuída, permitindo o processamento de dados em vários nós. O Catalyst Optimizer otimiza as consultas SQL, gerando planos de execução eficientes. O sistema de gerenciamento de memória Spark otimiza o uso da memória, enquanto o Tungsten Execution Engine melhora o desempenho ainda mais, usando a compilação Just-In-Time (JIT) para melhorar a velocidade de execução.

**Parágrafo 3:**

A fórmula geral para calcular o tempo de execução (RT) em um cluster Spark é:

RT = (Job Size / CPU Cores) x (1 + (Data Size / Network Bandwidth))

onde Job Size é o tamanho dos dados a serem processados, CPU Cores é o número de núcleos de CPU disponíveis, Data Size é o tamanho dos dados que precisam ser transferidos pela rede e Network Bandwidth é a largura de banda disponível.

Item do edital: Arquitetura de cloud computing para ciência de dados (AWS Azure GCP)

**Parágrafo 1:**

A arquitetura de cloud computing para ciência de dados é essencial para facilitar o armazenamento, processamento e análise eficientes de grandes conjuntos de dados. As principais plataformas de cloud, como AWS, Azure e GCP, oferecem serviços de computação escaláveis, armazenamento flexível e ferramentas de análise especializadas para atender às necessidades específicas da ciência de dados.

**Parágrafo 2:**

Essas plataformas de cloud fornecem uma gama de serviços essenciais, incluindo clusters de computação para processamento paralelo, armazenamento de objetos para conjuntos de dados volumosos, bancos de dados para gerenciamento de dados estruturados e ferramentas de aprendizado de máquina para modelagem preditiva. Os serviços de cloud podem ser integrados com ferramentas de ciência de dados de código aberto e proprietárias para criar pipelines completos de ciência de dados.

**Parágrafo 3:**

A arquitetura de cloud computing otimiza os recursos de computação, reduzindo custos e aumentando a eficiência. Ela permite que os cientistas de dados aloquem recursos dinamicamente com base nas cargas de trabalho, eliminem a necessidade de gerenciar infraestrutura física e colaborem facilmente com outros especialistas em um ambiente seguro e compartilhado.

Item do edital: Álgebra relacional e SQL (padrão ANSI)

**Álgebra Relacional**

A Álgebra Relacional é uma linguagem formal que usa operadores para manipular e consultar dados em bancos de dados relacionais. Os operadores fundamentais incluem seleção (σ), projeção (π), junção (⋈), união (∪), interseção (∩) e diferença (-). Esses operadores permitem que os usuários extraiam, combinem e processem dados de várias maneiras. Por exemplo, a consulta "selecione todos os nomes de alunos com notas maiores que 80" pode ser expressa como σ(gpa > 80)(alunos).

**SQL (padrão ANSI)**

A Linguagem de Consulta Estruturada (SQL) é uma linguagem de banco de dados padronizada que é amplamente usada para interagir com bancos de dados relacionais. O padrão ANSI SQL define um conjunto abrangente de comandos e recursos que permitem aos usuários criar, consultar, atualizar e manipular dados. O SQL é derivado de conceitos de álgebra relacional e fornece uma sintaxe fácil de usar que é acessível para usuários não técnicos.

**Comparação**

A Álgebra Relacional é uma linguagem teórica que fornece uma base matemática para operações de banco de dados. É mais concisa e abstrata do que o SQL. Por outro lado, o SQL é uma linguagem prática projetada para uso em sistemas de banco de dados do mundo real. Ele oferece uma gama mais ampla de recursos e funcionalidades, incluindo suporte a transações, controle de acesso e manipulação de dados.

Item do edital: SQL Server

O SQL Server é um sistema de gerenciamento de banco de dados relacional (SGBDR) desenvolvido pela Microsoft. Ele armazena e gerencia dados em um formato tabular estruturado, permitindo que os usuários acessem e manipulem dados de forma eficiente. O SQL Server usa a linguagem de consulta estruturada (SQL) para interagir com os dados, o que permite aos usuários consultar, inserir, atualizar e excluir registros de maneira declarativa.

O SQL Server é popular entre empresas devido à sua confiabilidade, escalabilidade e recursos robustos, como alta disponibilidade, recuperação de desastres e segurança avançada. Ele oferece vários recursos para otimizar o desempenho, como indexação, particionamento e armazenamento em memória (In-Memory OLTP).

A edição Standard do SQL Server usa o modelo de licenciamento baseado em núcleo. O custo de licenciamento é calculado multiplicando o número de núcleos do servidor pelo preço por núcleo. A fórmula para calcular o custo de licenciamento é: Custo de Licenciamento = Número de Núcleos \* Preço por Núcleo. Existem outras edições do SQL Server disponíveis, como Enterprise, Developer e Express, cada uma com recursos e opções de licenciamento diferentes.

Item do edital: PostgreSQL

PostgreSQL é um sistema de gerenciamento de banco de dados relacional de código aberto e gratuito, desenvolvido pela PostgreSQL Global Development Group. É conhecido por sua confiabilidade, escalabilidade e conjunto abrangente de recursos. Ele é compatível com a linguagem de consulta estruturada (SQL) e oferece suporte a transações ACID (Atomicity, Consistency, Isolation, Durability).

Uma característica fundamental do PostgreSQL é seu mecanismo de otimização de consultas, que utiliza um otimizador de consultas baseado em custo para gerar planos de execução eficientes. O otimizador leva em consideração fatores como estatísticas de tabela, índices e restrições para determinar o melhor plano de execução para cada consulta. A fórmula usada para estimar o custo de um plano de execução é:

**Custo = (Custo do CPU + Custo do I/O) \* Número de linhas**

Outra característica importante do PostgreSQL é sua arquitetura de processamento de consultas paralelas, que permite que as consultas sejam executadas em paralelo em vários processadores ou núcleos de CPU. Isso melhora significativamente o desempenho para consultas complexas ou grandes conjuntos de dados. O PostgreSQL também oferece suporte a particionamento de tabela, que permite dividir grandes tabelas em partes menores para melhorar ainda mais a escalabilidade e o desempenho.

Item do edital: MySQL

MySQL é um sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto amplamente usado, desenvolvido por Oracle Corporation. Ele armazena e gerencia dados usando uma estrutura baseada em tabela, onde os dados são organizados em linhas e colunas. O MySQL é conhecido por sua velocidade, confiabilidade e facilidade de uso, o que o torna popular para uma ampla gama de aplicativos, desde pequenas empresas até grandes organizações.

O MySQL usa a linguagem de consulta estruturada (SQL) para interagir com os dados. SQL é uma linguagem padronizada que permite aos usuários consultar, inserir, atualizar e excluir dados no banco de dados. O MySQL também suporta uma ampla gama de tipos de dados, incluindo números, strings, datas e blobs binários.

O MySQL é altamente escalonável e pode lidar com grandes volumes de dados. Ele também oferece uma variedade de recursos avançados, como replicação, particionamento e criptografia de dados. Além disso, o MySQL é compatível com uma ampla gama de plataformas de hardware e software, tornando-o uma escolha versátil para diversas implantações.

Item do edital: Bnco de dados NoSQL.

**Parágrafo 1:** Os Banco de Dados NoSQL (Not Only SQL) são sistemas de gerenciamento de dados projetados para manipular grandes quantidades de dados não estruturados ou semiestruturados. Esses dados não se encaixam bem nos modelos de dados relacionais tradicionais que usam tabelas e esquemas rígidos. O NoSQL oferece uma abordagem mais flexível e escalável, permitindo que os dados sejam armazenados em formatos como documentos, pares de valores-chave ou gráficos.

**Parágrafo 2:** Os bancos de dados NoSQL diferem dos bancos de dados relacionais em vários aspectos-chave. Eles geralmente são distribuídos, o que significa que os dados são armazenados em vários servidores. Isso proporciona alta disponibilidade e escalabilidade. Além disso, os bancos de dados NoSQL oferecem flexibilidade de esquema, permitindo que os dados sejam adicionados ou modificados dinamicamente sem alterar a estrutura do banco de dados. Isso é particularmente benéfico para dados de rápido crescimento e evolução.

**Parágrafo 3:** Existem vários modelos de dados NoSQL, cada um projetado para atender a requisitos específicos. Os modelos comuns incluem:

* **Documentos (JSON/XML):** Armazena dados em registros flexíveis e hierárquicos.
* **Pares de valores-chave:** Armazena dados como um conjunto de pares de chave-valor, fornecendo acesso rápido e simples.
* **Gráficos:** Armazena dados como nós e arestas interconectados, representando relacionamentos complexos.

Item do edital: Banco de dados e formatos de arquivo orientado a colunas

**Parágrafo 1:** Um banco de dados orientado a colunas armazena dados em colunas, em vez de linhas como em bancos de dados tradicionais orientados a linhas. Ele é otimizado para consultas analíticas que envolvem grandes quantidades de dados e agregações complexas. Os bancos de dados orientados a colunas geralmente usam fórmulas como soma (SUM), média (AVG) e contagem (COUNT) para processar os dados.

**Parágrafo 2:** Os formatos de arquivo orientados a colunas são projetados para armazenar dados em um layout otimizado para leitura e processamento eficientes. Eles são comumente usados para armazenar dados analíticos que precisam ser acessados ​​rapidamente para relatórios e análises. Um exemplo popular de um formato de arquivo orientado a colunas é o Apache Parquet, que usa compressão de dados para reduzir o tamanho do arquivo e melhorar o desempenho.

**Parágrafo 3:** Os bancos de dados e formatos de arquivo orientados a colunas oferecem vantagens significativas sobre seus equivalentes orientados a linhas para cenários de análise de dados. Eles permitem consultas mais rápidas, redução do tamanho dos arquivos e escalabilidade aprimorada. Além disso, eles são adequados para uma ampla gama de aplicativos, incluindo inteligência de negócios, aprendizado de máquina e análise preditiva.

Item do edital: Parquet

Parquet é um formato de arquivo de código aberto projetado para armazenar dados em colunas, tornando-o adequado para análise de big data e processamento paralelo. Utiliza um esquema de codificação que comprime cada coluna individualmente, resultando em tamanhos de arquivo menores e tempos de leitura mais rápidos.

O Parquet divide os dados em grupos de linhas chamados blocos de linha e, em seguida, codifica cada coluna dentro de um bloco de linha usando uma técnica específica. Para dados numéricos, o Parquet usa codificação delta-RLE (Range-Length Encoding), que armazena a diferença entre valores consecutivos e os comprime usando codificação run-length. Para dados categóricos, ele usa a codificação bit-packing, que agrupa valores semelhantes em bits e os compacta.

A fórmula para calcular a economia de espaço obtida usando a codificação delta-RLE é: (1 - (mean(delta) / mean(value))) \* 100, onde delta é a diferença entre valores consecutivos e value é o valor original. A economia de espaço para codificação bit-packing é calculada como (1 - (num\_bits\_used / num\_bits\_possible)) \* 100, onde num\_bits\_used é o número de bits usados para representar os valores e num\_bits\_possible é o número total de bits possíveis.

Item do edital: MonetDB

Sympathizert

Item do edital: duckDB.

DuckDB é um banco de dados analítico em memória projetado para máxima velocidade e facilidade de uso. Ele usa uma arquitetura de coluna, onde os dados são armazenados em colunas compactadas, permitindo consultas rápidas e eficientes. DuckDB também suporta consultas interativas por meio de um interpretador SQL incorporado, tornando-o conveniente para exploradores de dados e analistas.

DuckDB é notável por seu desempenho excepcional, alcançando velocidades de consulta até 100x mais rápidas que os bancos de dados tradicionais. Ele aproveita vários recursos de otimização, como compilação JIT (Just-In-Time), processamento de vetores e algoritmos especializados para consultas comuns. Isso o torna ideal para processamento de dados em tempo real, análise ad-hoc e cenários de aprendizado de máquina.

Além de seu desempenho, DuckDB é conhecido por sua simplicidade e facilidade de uso. Possui uma interface SQL padrão, o que significa que os usuários podem escrever consultas usando a sintaxe SQL familiar. DuckDB também oferece suporte a funções e extensões personalizadas, permitindo que os usuários estendam seus recursos para atender às necessidades específicas do aplicativo.

Item do edital: Normalização numérica.

A normalização numérica é um processo de transformação de dados numéricos em uma forma mais utilizável. Isso envolve converter os dados em um intervalo específico ou escala para melhorar a interpretabilidade, comparação e processamento dos dados.

A normalização pode ser realizada usando várias fórmulas, incluindo normalização Min-Max, normalização da pontuação Z e normalização decimal. A normalização Min-Max mapeia os dados para o intervalo [0, 1] usando a fórmula: (x - min) / (max - min), onde x é o valor original, min é o valor mínimo e max é o valor máximo. A normalização da pontuação Z padroniza os dados com uma média de 0 e um desvio padrão de 1, usando a fórmula: (x - μ) / σ, onde x é o valor original, μ é a média e σ é o desvio padrão. A normalização decimal converte os dados em um formato decimal fixo, geralmente com duas casas decimais, truncado ou arredondado conforme necessário.

A normalização é amplamente utilizada em vários campos, como aprendizado de máquina, processamento de sinais e análise estatística. Permite melhor comparação de dados de diferentes escalas, melhora a precisão dos algoritmos e facilita a interpretação dos resultados.

Item do edital: Discretização.

**Parágrafo 1:**

A discretização é um processo matemático que envolve dividir um objeto ou conceito contínuo em unidades distintas ou discretas. Isso simplifica objetos complexos, permitindo que sejam representados por meio de valores numéricos discretos. Por exemplo, um sinal analógico contínuo pode ser discretizado em valores digitais por meio da amostragem.

**Parágrafo 2:**

Existem diferentes métodos de discretização, dependendo do objeto ou conceito sendo discretizado. Um método comum é a amostragem, que divide um objeto contínuo em intervalos iguais e atribui valores numéricos a cada intervalo. Outros métodos incluem quantização, que divide um intervalo contínuo em níveis discretos, e pixelização, que divide uma imagem em pixels individuais.

**Parágrafo 3:**

A discretização é fundamental em vários campos, incluindo processamento de sinais, processamento de imagens, computação gráfica e simulação numérica. Na computação, os valores discretos são armazenados em memória e processados usando algoritmos específicos. A discretização também tem aplicações em áreas como física, engenharia e finanças, onde permite que conceitos complexos sejam modelados e analisados usando computadores.

Item do edital: Tratamento de dados ausentes.

**Tratamento de Dados Ausentes**

Dados ausentes são um problema comum na análise de dados, podendo comprometer a precisão e a validade das conclusões. Existem várias abordagens para lidar com dados ausentes, cada uma com suas próprias vantagens e desvantagens.

**Métodos de Imputação:**

Um método comum é imputação, que envolve preencher os valores ausentes com valores estimados. A imputação por média ou mediana substitui os valores ausentes pela média ou mediana da variável, respectivamente. A imputação por moda substitui os valores ausentes pelo valor mais frequente. Métodos estatísticos mais avançados, como imputação múltipla por cadeia de Markov (MICE), podem levar em conta a relação entre as variáveis para imputar valores mais precisos.

**Exclusão de Casos:**

Outra abordagem é excluir casos com valores ausentes. Isso pode ser apropriado quando os dados ausentes são aleatórios e não relacionados a outras variáveis. No entanto, a exclusão de casos pode levar à perda de informações valiosas e potencializar o viés de seleção.

**Modelagem de Dados Ausentes:**

Métodos estatísticos como regressão múltipla ou modelos de equações estruturais podem ser usados para modelar os dados ausentes. Esses métodos reconhecem a presença de dados ausentes e incorporam a incerteza associada às estimativas. A modelagem de dados ausentes pode fornecer resultados mais precisos do que a imputação ou exclusão de casos.

Item do edital: Tratamento de outliers e agregações.

**Tratamento de Outliers**

Outliers são valores extremos que podem distorcer os resultados de análises estatísticas. Eles podem ser tratados removendo-os ou modificando-os para torná-los mais consistentes com o restante do conjunto de dados. Uma técnica comum é usar o desvio absoluto médio (MAD), que remove os valores que estão mais de 2,5 MADs da mediana.

**Agregações**

As agregações combinam vários valores em um único valor, como média, mediana ou desvio padrão. Elas podem ser usadas para resumir grandes conjuntos de dados e torná-los mais gerenciáveis. A fórmula para a média é:

Média = (x1 + x2 + ... + xn) / n

onde x1, x2, ..., xn são os valores no conjunto de dados e n é o número de valores.

**Conclusão**

O tratamento de outliers e as agregações são técnicas valiosas para limpar e preparar dados para análise. Os outliers podem ser identificados e removidos ou modificados para reduzir sua influência nos resultados. As agregações podem resumir grandes conjuntos de dados em valores únicos, facilitando a interpretação e a análise.

Item do edital: Tratamento de dados: Matching

**Parágrafo 1:**

O Matching, também conhecido como Emparelhamento Ótimo, é uma técnica de otimização que determina os pares ideais em um conjunto de objetos com base em critérios pré-especificados. Ele busca encontrar um conjunto de pares que maximizem ou minimizem uma função objetivo. Por exemplo, em um mercado de trabalho, o Matching pode ser usado para atribuir candidatos a vagas para maximizar a satisfação do empregador e do candidato.

**Parágrafo 2:**

Existem vários algoritmos de Matching, cada um com seus próprios pontos fortes e fracos. Um algoritmo comum é o Hungarian Algorithm, que tem complexidade de tempo O(n³), onde n é o número de objetos. Este algoritmo encontra o Matching de peso máximo em um grafo bipartido ponderado. Para grafos mais complexos, outros algoritmos, como o Matching via Fluxos de Rede, podem ser mais eficientes.

**Parágrafo 3:**

O Matching tem aplicações diversas em áreas como:

* **Alocação de Recursos:** Determinar a melhor alocação de recursos limitados entre várias tarefas ou indivíduos.
* **Agendamento:** Criar horários que otimizem o uso de recursos e minimizem conflitos.
* **Logística:** Encontrar rotas ou sequências de atividades que minimizem tempo ou custos.

Item do edital: Deduplicação.

**Parágrafo 1:**

A dedução é um processo de remover dados redundantes de um conjunto de dados, preservando ao mesmo tempo sua integridade. Seu objetivo é eliminar dados duplicados que podem ocorrer devido a erros de entrada, integração de dados de várias fontes ou atualizações inconsistentes. Ao remover registros duplicados, a dedução melhora a precisão e a eficiência dos dados, facilitando a análise, o gerenciamento e o compartilhamento.

**Parágrafo 2:**

Existem vários algoritmos e técnicas usadas na dedução, incluindo:

* **Algoritmo de Sorteen (Fórmula):**

Σ = {t / t ∈ T} para todas as tabelas T no banco de dados

L = {f / f é um atributo de cada t ∈ Σ}

K = {k / k é uma chave candidata de cada t ∈ Σ}

* **Algoritmo de Bloqueio:** Agrupa registros em blocos com base em valores de atributo semelhantes, reduzindo o número de comparações necessárias.
* **Algoritmo de Histogramas:** Cria um histograma para distribuir os valores dos atributos, identificando valores duplicados em potencial com alta frequência.

**Parágrafo 3:**

A dedução oferece vários benefícios, incluindo:

* **Otimização do armazenamento:** Remoção de dados duplicados reduz o espaço de armazenamento necessário.
* **Melhoria de desempenho:** A eliminação de registros desnecessários acelera as consultas e transações.
* **Aumento da precisão:** A remoção de dados duplicados elimina erros e inconsistências, melhorando a confiabilidade dos dados.
* **Simplificação da análise:** Conjuntos de dados mais limpos e compactos são mais fáceis de analisar e interpretar.

Item do edital: Data cleansing.

**Data Cleansing: Resumo em Três Parágrafos**

Data cleansing é o processo de identificar e corrigir dados inconsistentes, imprecisos ou ausentes em um conjunto de dados. Isso envolve uma série de técnicas para remover ou corrigir valores duplicados, lidar com valores ausentes, reformatar dados e validar a precisão dos dados. O objetivo da data cleansing é garantir que os dados sejam confiáveis e precisos para análise e tomada de decisão.

Uma etapa crucial na data cleansing é a identificação de valores ausentes, que podem ser indicados por valores em branco ou por códigos específicos (por exemplo, NULL ou NA). Os valores ausentes podem ser tratados de várias maneiras, como removendo linhas ou colunas com muitos valores ausentes ou imputando valores com base em valores vizinhos ou médias. O método de imputação usado dependerá do tipo de dados e da natureza dos valores ausentes.

Além dos valores ausentes, a data cleansing também aborda dados inconsistentes ou incorretos. Isso pode incluir valores duplicados, valores fora do intervalo esperado ou erros de digitação. Para corrigir essas inconsistências, podem ser usadas fórmulas como a função PROCV (por exemplo, **=PROCV(A2; Tabela1!A:B; 2; FALSO)**) para substituir valores incorretos por valores corretos ou a função SE (por exemplo, **=SE(C2="";"";C2)**) para remover valores em branco.

Item do edital: Enriquecimento de dados.

**Parágrafo 1:** O enriquecimento de dados é o processo de aprimorar conjuntos de dados adicionando informações contextuais e relevantes de várias fontes. Isso melhora a qualidade e o valor dos dados, permitindo análises mais aprofundadas e tomadas de decisão aprimoradas. O enriquecimento de dados ajuda a preencher lacunas, corrigir erros e adicionar novos atributos aos dados existentes.

**Parágrafo 2:** Existem vários tipos de enriquecimento de dados, incluindo enriquecimento interno (usando dados de dentro da organização) e enriquecimento externo (usando dados de fontes externas). As técnicas comuns de enriquecimento de dados incluem correspondência, fusão e agregação. A correspondência vincula registros de diferentes conjuntos de dados com base em identificadores comuns. A fusão combina informações de vários registros para criar um único registro mais abrangente. A agregação resume ou agrupa dados para identificar padrões e tendências.

**Parágrafo 3:** O enriquecimento de dados pode ter um impacto significativo nas organizações, pois:

* **Melhora a tomada de decisão:** Dados aprimorados fornecem insights mais profundos, levando a decisões mais informadas e melhores resultados.
* **Personaliza experiências do cliente:** Dados enriquecidos permitem que as empresas entendam melhor seus clientes e personalizem mensagens, ofertas e experiências.
* **Otimiza operações:** O enriquecimento de dados identifica ineficiências, otimiza processos e melhora a eficiência geral.

Fórmulas comumente usadas no enriquecimento de dados:

* **Coeficiente de Jaccard:** Mede a similaridade entre dois conjuntos de dados (J(A, B) = |A ∩ B| / |A ∪ B|)
* **Distância de Levenshtein:** Calcula a distância de edição entre duas sequências de caracteres (d(A, B) = número mínimo de inserções, exclusões ou substituições necessárias para transformar A em B)

Item do edital: Desidentificação de dados sensíveis.

**Parágrafo 1:**

A desidentificação de dados sensíveis remove informações pessoais identificáveis (PII) de conjuntos de dados, mantendo simultaneamente a utilidade dos dados para análise e pesquisa. Isso envolve a substituição de PII por dados anônimos ou pseudoanônimos, garantindo que os indivíduos não possam ser reidentificados. O objetivo é proteger a privacidade dos indivíduos, permitindo que os dados sejam usados com segurança.

**Parágrafo 2:**

Existem vários métodos de desidentificação de dados, incluindo supressão, mascaramento, tokenização e criptografia. A supressão envolve a remoção de dados identificáveis, enquanto o mascaramento substitui esses dados por valores fictícios. A tokenização substitui PII por tokens exclusivos, vinculando os tokens aos dados originais por meio de uma tabela de pesquisa segura. A criptografia protege os dados identificáveis tornando-os ilegíveis sem uma chave de descriptografia.

**Parágrafo 3:**

A fórmula para calcular o risco de reidentificação após a desidentificação é:

**Risco de reidentificação = P(reidentificação) \* Custo (reidentificação)**

onde:

* P(reidentificação) é a probabilidade de um indivíduo ser reidentificado com base nos dados desidentificados
* Custo(reidentificação) é o impacto potencial da reidentificação no indivíduo

Item do edital: Algoritmos fuzzy matching

**Parágrafo 1:**

Os algoritmos de correspondência aproximada são técnicas computacionais projetadas para identificar semelhanças entre strings, mesmo quando essas strings contêm erros ou variações. Ao contrário dos algoritmos de correspondência exata, que exigem uma correspondência perfeita, os algoritmos de correspondência aproximada permitem um certo grau de imprecisão. Eles são usados em diversos aplicativos, como pesquisa de texto, reconhecimento de fala e processamento de linguagem natural.

**Parágrafo 2:**

Existem vários algoritmos de correspondência aproximada, cada um com seus próprios pontos fortes e fracos. Um algoritmo comumente usado é o Levenshtein, que mede a distância de edição entre duas strings, ou seja, o número mínimo de inserções, exclusões ou substituições necessárias para transformar uma string na outra. A fórmula do Levenshtein é:

d(s1, s2) = min( d(s1[:-1], s2) + 1,

d(s1, s2[:-1]) + 1,

d(s1[:-1], s2[:-1]) + (s1[-1] != s2[-1]) )

onde s1 e s2 são as duas strings e d(s1, s2) é a distância de edição entre elas.

**Parágrafo 3:**

Outros algoritmos de correspondência aproximada incluem o Jaro-Winkler, que considera transposições de caracteres, e o Sørensen-Dice, que é adequado para strings com comprimentos muito diferentes. Esses algoritmos atribuem um valor de semelhança entre 0 e 1, onde 1 indica uma correspondência perfeita e 0 indica nenhuma semelhança. Ao escolher um algoritmo de correspondência aproximada, é importante considerar fatores como a precisão, eficiência e tolerância a erros que são relevantes para a aplicação específica.

Item do edital: Algoritmos stemming.

Os algoritmos stemming são técnicas de processamento de linguagem natural que visam reduzir palavras a suas raízes ou radicais. Eles desempenham um papel crucial em várias aplicações de PNL, como recuperação de informação, categorização de texto e análise de sentimentos. O objetivo principal do stemming é remover sufixos e prefixos comuns das palavras, resultando em representações mais abstratas e genéricas.

Existem vários algoritmos stemming disponíveis, cada um com suas peculiaridades. Um algoritmo stemming popular é o Stemming de Porter, que usa uma série de regras para remover sufixos comuns da língua inglesa. Por exemplo, a regra de remoção de sufixo "ional" remove o sufixo "-ional" de palavras como "nacional" e "promocional", resultando no radical "nacion" e "promoc".

Outro algoritmo stemming amplamente utilizado é o Stemming de Lancaster, que emprega um dicionário e uma série de regras para identificar e remover sufixos. Ao contrário do Stemming de Porter, o Stemming de Lancaster preserva alguns sufixos que podem ser significativos em certas aplicações. A fórmula geral para o Stemming de Lancaster pode ser representada como: ($ Palavra\_stem = Palavra\_original(raiz) + Ajuste(raiz) + Desinência(raiz)) , onde:

* ($ Palavra\_stem ) é a palavra reduzida
* ($ Palavra\_original ) é a palavra original
* ($ raiz ) é a raiz da palavra
* ($ Ajuste ) é um ajuste opcional aplicado à raiz
* ($ Desinência ) é uma desinência opcional adicionada à raiz

Item do edital: Visualização e análise exploratória de dados.

**Parágrafo 1:**

A Visualização de Dados é a representação gráfica de informações para facilitar a compreensão e identificação de padrões. Ela ajuda a explorar e analisar dados rapidamente, fornecendo uma visão geral visual. Métodos comuns incluem gráficos de barras, gráficos de pizza e gráficos de dispersão.

**Parágrafo 2:**

A Análise Exploratória de Dados (AED) é um processo iterativo que envolve a exploração, visualização e modelagem de dados para descobrir padrões, identificar outliers e gerar hipóteses. A AED ajuda a obter uma compreensão abrangente dos dados e a tomar decisões informadas. Técnicas comuns incluem análise de frequência, medidas de tendência central (por exemplo, média, mediana) e desvio padrão.

**Parágrafo 3:**

Juntas, a Visualização de Dados e a AED permitem uma compreensão profunda dos dados. Ao visualizar os dados, podemos identificar padrões visuais e outliers. Ao analisar esses padrões usando técnicas de AED, podemos Quantificar padrões, testar hipóteses e desenvolver modelos preditivos. Esta combinação poderosa de técnicas ajuda na tomada de decisões baseada em dados e na resolução de problemas.

Item do edital: Linguagem de programação R.

A Linguagem de Programação R é uma linguagem de programação de código aberto e gratuita usada principalmente para análise estatística, manipulação de dados e gráficos. Foi desenvolvida no final da década de 1990 por estatísticos e cientistas da computação na Universidade de Auckland e lançada em 1993.

A linguagem R é baseada em S, uma linguagem de programação estatística desenvolvida nos laboratórios Bell na década de 1980. R herdou a sintaxe e funções de S, mas também introduziu vários recursos novos, incluindo um sistema de tipagem mais rigoroso, um sistema de funções aninhadas e uma interface de programação de aplicações (API) mais ampla.

R é uma linguagem interpretada, o que significa que os comandos são executados linha por linha. Possui uma variedade de recursos estatísticos e gráficos, incluindo funções para análise descritiva, inferência estatística, modelagem e visualização de dados. R também tem uma comunidade ativa de usuários e contribuidores, que desenvolvem e mantêm pacotes de software adicionais, aumentando ainda mais suas capacidades.

Item do edital: Linguagem de programação Python.

Python é uma linguagem de programação interpretada, de alto nível e de uso geral. É conhecida por sua sintaxe simples e fácil de ler, tornando-a popular entre iniciantes e desenvolvedores experientes. O Python adota uma abordagem orientada a objetos e oferece suporte a programação funcional, procedimental e lógica.

O Python é uma linguagem dinamicamente tipada, o que significa que os tipos de dados das variáveis são verificados em tempo de execução. Isso permite maior flexibilidade, mas pode levar a erros se os tipos forem inconsistentes. O Python usa um sistema de gerenciamento de memória de contagem de referências que libera automaticamente objetos não utilizados.

O Python é amplamente utilizado para desenvolvimento web, aprendizado de máquina, ciência de dados e automação de tarefas. Sua ampla gama de bibliotecas e frameworks torna fácil a integração com outros serviços e sistemas. A sintaxe do Python é concisa e usa recuo em vez de chaves para agrupar blocos de código, o que contribui para sua legibilidade e manutenibilidade. A linguagem também oferece suporte a programação paralela e concorrente por meio do módulo "threading".

Item do edital: Linguagem de programação Scala.

A linguagem de programação Scala é uma linguagem de propósito geral que combina programação orientada a objetos e programação funcional. Ela é projetada para ser concisa, expressiva e escalável, oferecendo recursos como classes de casos, correspondência de padrões e tipos inferidos.

Scala é uma linguagem compilada que roda na Máquina Virtual Java (JVM) ou em outras plataformas que suportam a JVM. Isso permite que os programas Scala sejam executados em vários sistemas operacionais e dispositivos. A sintaxe de Scala é semelhante ao Java, mas com adições que facilitam a programação funcional e a segurança do tipo.

Os recursos de programação funcional de Scala incluem funções de alta ordem, imutabilidade e recursão de cauda. A linguagem também possui um poderoso sistema de tipos que ajuda a detectar e prevenir erros em tempo de compilação. A combinação de recursos de programação orientada a objetos e funcional torna a Scala adequada para uma ampla gama de aplicações, incluindo processamento de dados, desenvolvimento web e aprendizado de máquina.

Item do edital: Programação funcional.

A programação funcional é um paradigma de programação que enfatiza a avaliação de expressões matemáticas puras e o uso de funções como valores de primeira classe. Em vez de alterar o estado de variáveis, as funções funcionais criam novos valores sem efeitos colaterais. Elas também seguem o princípio da transparência referencial, ou seja, o valor de uma expressão depende apenas de seus argumentos de entrada.

As funções funcionais são frequentemente compostas usando técnicas como aplicação de funções, filtragem e mapeamento. A aplicação de funções envolve passar o resultado de uma função como argumento para outra, enquanto a filtragem remove elementos de uma lista que não atendem a um determinado critério e o mapeamento aplica uma função a cada elemento de uma lista. Essas técnicas permitem a construção de programas complexos a partir de blocos de construção mais simples.

Um exemplo clássico da programação funcional é a função fold, que reduz uma lista de valores a um único valor, aplicando repetidamente uma função binária aos pares de elementos adjacentes da lista. A fórmula para fold é:

fold(f, init, [x1, x2, ..., xn]) = f(f(f(...(f(init, x1), x2), ...), xn)

Item do edital: Programação orientada a objetos.

A programação orientada a objetos é um paradigma de programação que organiza o código em torno de objetos, que possuem dados (atributos) e comportamentos (métodos). Cada objeto é uma instância de uma classe, que define sua estrutura e comportamento. As classes podem herdar umas das outras, criando hierarquias de classes que permitem a reutilização de código.

O princípio fundamental da programação orientada a objetos é o encapsulamento, que mantém dados e comportamentos juntos em um único objeto. Isso promove a ocultação da implementação, tornando o código mais fácil de manter e reutilizar. Além disso, objetos podem interagir entre si através do envio de mensagens, que invocam métodos em outros objetos.

A coesão e a separação de preocupações são benefícios importantes da programação orientada a objetos. Cada objeto é responsável por uma tarefa específica, e as interações entre os objetos são definidas claramente, tornando o código mais fácil de gerenciar e depurar. Além disso, o polimorfismo permite que objetos de classes diferentes respondam de forma diferente às mesmas mensagens, promovendo a flexibilidade e a reutilização de código.

Item do edital: Classes de objetos e suas propriedades (vetores listas data frames).

**Classes de Objetos**

As classes de objetos são as estruturas de dados básicas usadas para armazenar e manipular dados em R. As classes mais comuns incluem:

* **Vetores:** Vetores são sequências unidimensionais de elementos do mesmo tipo. Eles podem armazenar números, strings, fatores ou valores lógicos.
* **Listas:** As listas são estruturas de dados flexíveis que podem conter uma variedade de elementos, incluindo vetores, matrizes, data frames e outros objetos R.
* **Data Frames:** Os data frames são tabelas organizadas em linhas e colunas. Cada coluna contém dados de um tipo específico, como números, strings ou fatores.

**Propriedades de Objetos**

Cada classe de objeto possui um conjunto de propriedades que definem seus atributos. As propriedades comuns incluem:

* **Comprimento:** O comprimento do objeto, que é o número de elementos que ele contém.
* **Tipo de dados:** O tipo de dados dos elementos do objeto, como numérico, string ou lógico.
* **Nome das colunas:** Para data frames, os nomes das colunas que identificam cada variável.

**Fórmulas**

Algumas propriedades de objetos podem ser calculadas usando fórmulas, como:

* **Comprimento de um vetor:** length(vetor)
* **Tipo de dados de um objeto:** typeof(objeto)

Item do edital: Manipulação e tabulação de dados com numpy

**Manipulação de Dados:**

O NumPy fornece funções robustas para manipular dados em matrizes multidimensionais. Você pode executar operações aritméticas (p. ex., +, \*), lógicas (p. ex., &, |) e relacionais (p. ex., >, <) em matrizes inteiras, resultando em novas matrizes com os valores calculados. Além disso, o fatiamento e a indexação permitem extrair subconjuntos específicos de dados de matrizes.

**Tabulação de Dados:**

O NumPy simplifica a tabulação de dados com funções como argmax() e argmin(), que retornam os índices dos valores máximo e mínimo em uma matriz, respectivamente. Para calcular estatísticas básicas como média, desvio padrão e variância, você pode usar funções como mean(), std() e var(). Essas funções retornam valores escalares que representam as estatísticas da matriz inteira.

**Fórmulas:**

* **Média (μ):** μ = (1/n) \* Σxi
* **Desvio Padrão (σ):** σ = √[(1/n) \* Σ(xi - μ)²]
* **Variância (σ²):** σ² = (1/n) \* Σ(xi - μ)²

Item do edital: Manipulação e tabulação de dados com pandas

**Parágrafo 1:** O Pandas é uma biblioteca Python essencial para manipulação e tabulação de dados. Ele fornece estruturas de dados flexíveis, chamadas DataFrames, que organizam e armazenam dados como tabelas com linhas e colunas. Os DataFrames permitem operações como seleção, filtragem, ordenação e agregação de dados. Para selecionar dados específicos, os índices Pandas podem ser usados para acessar linhas e colunas específicas. Por exemplo, df[linhas, colunas] seleciona um subconjunto de linhas e colunas do DataFrame df.

**Parágrafo 2:** Além da manipulação, o Pandas oferece recursos poderosos para tabulação de dados. O método groupby() agrupa dados com base em uma ou mais colunas e permite operações de agregação como mean(), sum() e std(). Por exemplo, df.groupby('coluna').mean() calcula a média das colunas agrupadas por 'coluna'. O Pandas também suporta operações de junção, como merge() e join(), que combinam DataFrames com base em colunas comuns. A junção interna (df1.merge(df2, on='chave')) combina linhas de df1 e df2 que possuem a mesma chave, enquanto a junção externa esquerda (df1.merge(df2, on='chave', how='left')) mantém todas as linhas de df1 e adiciona dados correspondentes de df2 quando disponível.

**Parágrafo 3:** O Pandas oferece uma interface intuitiva e expressiva para manipulação e tabulação de dados. Ele simplifica tarefas complexas, reduzindo a necessidade de loops e instruções condicionais. Com sua ampla gama de funcionalidades, o Pandas é uma ferramenta essencial para trabalhar com dados em Python, permitindo aos usuários extrair insights valiosos e tomar decisões informadas.

Item do edital: Manipulação e tabulação de dados com tidyverse

**Parágrafo 1**

O tidyverse é um conjunto abrangente de pacotes R para manipulação e tabulação de dados. Ele fornece uma estrutura consistente de funções que operam em data frames "tidy", onde cada linha representa uma observação e cada coluna representa uma variável. A função select() seleciona colunas específicas, enquanto a função filter() seleciona linhas com base em critérios lógicos (por exemplo, filter(valor > 5)). A função mutate() cria novas colunas a partir de dados existentes ou operações aritméticas (por exemplo, mutate(nova\_coluna = valor + 5)).

**Parágrafo 2**

O tidyverse inclui funções para resumo e visualização de dados. A função summarise() calcula estatísticas agregadas, como soma (por exemplo, summarise(soma(valor))). A função group\_by() agrupa dados por valores específicos (por exemplo, group\_by(grupo)) antes de aplicar funções de resumo. Funções de visualização como ggplot() permitem criar gráficos flexíveis e informativos, onde cada ponto de dados é representado como um "geom" (por exemplo, geom\_line() para linhas ou geom\_bar() para barras).

**Parágrafo 3**

O tidyverse enfatiza a consistência e a legibilidade do código. Ele usa verbos (por exemplo, select() e filter()), em vez de operadores de atribuição (por exemplo, <-), para melhorar a compreensão. O tidyverse também promove a "programação funcional", onde os dados são transformados por meio de uma série de operações discretas, tornando o código mais fácil de ler, depurar e reutilizar.

Item do edital: Manipulação e tabulação de dados com data.table

O data.table é uma estrutura de dados R altamente eficiente projetada para manipulação e tabulação de dados. Ele é baseado na estrutura de dados data.frame, mas oferece desempenho aprimorado e uma sintaxe mais intuitiva. Uma das principais vantagens do data.table é sua capacidade de executar operações de agregação (por exemplo, soma, média) e transformações (por exemplo, criar novas colunas, filtrar linhas) de forma eficiente usando sua sintaxe de "i". Por exemplo, para calcular a média de uma coluna chamada "valor" para cada valor único na coluna "grupo", você pode usar a seguinte fórmula:

data.table[ , mean(valor) , by = grupo]

Além disso, o data.table fornece recursos avançados para tabulação, como a função melt() que converte dados de formato amplo para formato longo e vice-versa. Isso permite fácil agregação e tabulação de dados. Por exemplo, para converter dados de formato amplo em formato longo, você pode usar a seguinte fórmula:

data.table <- melt(data.table, id.vars = c("id", "grupo"))

Item do edital: Visualização de dados com ggplot

O ggplot é uma biblioteca R poderosa para visualização de dados. Baseia-se no conceito de mapeamentos estéticos, onde as variáveis ​​de dados são mapeadas para propriedades visuais, como cor, forma e tamanho. Isso permite que os usuários criem representações gráficas flexíveis e personalizadas de seus dados.

O ggplot usa uma sintaxe gramatical, que se assemelha à linguagem humana e torna a criação de gráficos fácil e intuitiva. A fórmula central do ggplot é **geom\_** + **aes**(), onde **geom\_** especifica o tipo de geometria (por exemplo, pontos, linhas ou barras) e **aes**() define os mapeamentos estéticos. Por exemplo, para criar um gráfico de dispersão, a fórmula seria ggplot(data, aes(x = var1, y = var2)) + geom\_point().

O ggplot oferece uma ampla gama de opções de personalização, incluindo controle sobre cores, legendas, eixos e outros elementos visuais. Ele também pode ser integrado a outras bibliotecas R, como dplyr e tidyr, para uma manipulação e preparação de dados mais abrangentes.

Item do edital: Visualização de dados com matplotlib.

Matplotlib é uma biblioteca Python amplamente utilizada para visualização de dados. Ele fornece uma ampla gama de opções de plotagem, incluindo gráficos de linha, gráficos de barras, gráficos de dispersão e gráficos de pizza. A biblioteca também oferece suporte a recursos avançados, como legendas, títulos e eixos personalizados.

Para criar um gráfico básico com Matplotlib, você pode usar a função plt.plot(). A função leva uma lista ou array de valores para o eixo x e uma lista ou array de valores para o eixo y. Por exemplo, para plotar uma linha reta, você pode usar o seguinte código:

import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]

y = [2, 4, 6, 8, 10]

plt.plot(x, y)

plt.show()

Este código irá plotar uma linha reta com uma inclinação de 2 e um intercepto y de 0. Você também pode usar Matplotlib para criar gráficos mais complexos, como gráficos de barras e gráficos de dispersão.

Item do edital: Paralelização de rotinas de ciência de dados.

**Parágrafo 1:** A paralelização na ciência de dados envolve a divisão de tarefas computacionalmente intensivas em subtarefas menores que podem ser executadas simultaneamente em vários processadores. Essa técnica melhora significativamente a eficiência e reduz o tempo de execução, especialmente para conjuntos de dados grandes ou operações complexas.

**Parágrafo 2:** Existem várias abordagens para paralelização, incluindo:

* **Paralelização de dados:** Divide os dados em blocos menores e os processa simultaneamente. Por exemplo, a fórmula para média de dados paralelizados é:

μ = (1/n) ∑[i=1:n] xi

* **Paralelização de tarefas:** Divide a tarefa em subtarefas independentes e as executa simultaneamente. Por exemplo, o tempo de execução de uma tarefa paralelizada com p processadores é dado por:

T = T0 / p

**Parágrafo 3:** A paralelização é especialmente benéfica para algoritmos iterativos, como treinamento de modelos de aprendizado de máquina, processamento de linguagem natural e otimização. Ao distribuir as iterações entre vários processadores, a convergência é acelerada e os resultados são obtidos mais rapidamente. No entanto, é importante observar que a sobrecarga de comunicação e sincronização pode afetar o desempenho da paralelização, portanto, a escolha da abordagem certa depende da natureza da tarefa e dos recursos computacionais disponíveis.

Item do edital: Probabilidade e probabilidade condicional.

**Probabilidade:** A probabilidade é uma medida numérica que representa a probabilidade de um evento ocorrer. É expressa como um número entre 0 e 1, onde 0 indica impossibilidade e 1 indica certeza. A probabilidade de um evento A é denotada por P(A). Por exemplo, se uma moeda é lançada e há duas possibilidades (cara ou coroa), cada uma delas tem uma probabilidade de 0,5.

**Probabilidade Condicional:** A probabilidade condicional é a probabilidade de um evento A ocorrer, dado que outro evento B já ocorreu. É denotada por P(A|B). Ela é calculada dividindo a probabilidade da ocorrência conjunta de A e B pela probabilidade de ocorrência de B:

P(A|B) = P(A e B) / P(B)

**Exemplo:** Suponha que uma caixa contém 10 bolas, sendo 5 vermelhas e 5 azuis. Se uma bola é retirada aleatoriamente da caixa e é vermelha, a probabilidade condicional de que a próxima bola retirada seja azul é:

P(Azul | Vermelha) = P(Azul e Vermelha) / P(Vermelha) = 5/10 = 0,5

Isso significa que, mesmo que a primeira bola retirada tenha sido vermelha, ainda há 50% de chance de que a próxima bola retirada seja azul.

Item do edital: Independência de eventos

**Parágrafo 1:**

Em teoria da probabilidade, eventos são considerados independentes quando a ocorrência de um não afeta a probabilidade de ocorrência de qualquer outro. Em outras palavras, a probabilidade de um evento acontecer permanece a mesma, independentemente de outro evento ter acontecido ou não.

**Parágrafo 2:**

Seja P(A) a probabilidade do evento A e P(B) a probabilidade do evento B. Os eventos A e B são independentes se a probabilidade da ocorrência conjunta de A e B for igual ao produto de suas probabilidades individuais:

P(A∩B) = P(A) × P(B)

**Parágrafo 3:**

A independência de eventos pode ser estendida para mais de dois eventos. Seja P(A₁, A₂, ..., Aₙ) a probabilidade da ocorrência conjunta de n eventos A₁, A₂, ..., Aₙ. Esses eventos são independentes se:

P(A₁∩A₂∩...∩Aₙ) = P(A₁) × P(A₂) × ... × P(Aₙ)

Item do edital: teorema de Bayes

**Parágrafo 1:**

O Teorema de Bayes é uma equação que relaciona a probabilidade condicional de dois eventos, P(A|B) e P(B|A), com suas probabilidades marginais, P(A) e P(B). Ele afirma que:

P(A|B) = (P(B|A) \* P(A)) / P(B)

**Parágrafo 2:**

Esta equação pode ser usada para atualizar as probabilidades à luz de novas informações. Por exemplo, se sabemos a probabilidade de um paciente ter uma doença (P(A)) e a probabilidade de um teste ser positivo para a doença se o paciente a tiver (P(B|A)), podemos calcular a probabilidade de o paciente ter a doença se o teste for positivo (P(A|B)).

**Parágrafo 3:**

O Teorema de Bayes tem ampla aplicação em vários campos, incluindo inteligência artificial, tomada de decisão e inferência estatística. Ele permite que atualizemos as crenças à medida que novas evidências são recebidas e é uma ferramenta fundamental para raciocínio probabilístico em situações de incerteza.

Item do edital: teorema da probabilidade total.

**Teorema da Probabilidade Total**

O Teorema da Probabilidade Total estabelece que a probabilidade de um evento composto, ou seja, um evento que pode ocorrer por meio de vários caminhos mutuamente exclusivos, é igual à soma das probabilidades de cada caminho individual.

**Fórmula:**

P(A) = P(A|B1)P(B1) + P(A|B2)P(B2) + ...

Onde:

* P(A) é a probabilidade do evento composto A
* P(A|Bi) é a probabilidade condicional de A dado que Bi ocorre
* P(Bi) é a probabilidade de Bi

**Interpretação:**

Para um evento composto com n caminhos mutuamente exclusivos, o teorema afirma que a probabilidade do evento composto é a soma das probabilidades de cada caminho individual. A probabilidade condicional de A dado que Bi ocorre representa a probabilidade de A ocorrer sob a condição de que Bi tenha ocorrido. A probabilidade de Bi representa a probabilidade de Bi ocorrer.

Item do edital: teorema da probabilidade total.

**Teorema da Probabilidade Total (PT)**

O Teorema da PT afirma que se um evento A pode ocorrer de n maneiras mutuamente exclusivas, e se B1, B2, ..., Bn são eventos que formam uma partição do espaço amostral, então a probabilidade de A é igual à soma das probabilidades de A ocorrer em cada um dos eventos B1, B2, ..., Bn.

Matematicamente, isso pode ser expresso como:

P(A) = ∑[P(A|B1) \* P(B1)] + ∑[P(A|B2) \* P(B2)] + ... + ∑[P(A|Bn) \* P(Bn)]

**Interpretação**

O Teorema da PT é útil para calcular a probabilidade de um evento que pode ocorrer de várias maneiras distintas. Ele divide o espaço amostral em subconjuntos menores e mutuamente exclusivos, calcula a probabilidade do evento em cada subconjunto e, em seguida, soma essas probabilidades para obter a probabilidade geral do evento.

**Aplicações**

O Teorema da PT tem diversas aplicações práticas, como:

* Calcular a probabilidade de um resultado específico em um experimento.
* Avaliar o risco de um evento ocorrer em cenários complexos.
* Fazer inferências estatísticas a partir de dados amostrais. Item do edital: Variáveis aleatórias e funções de probabilidade.

**Parágrafo 1:**

Variáveis aleatórias são quantidades que assumem valores de forma aleatória ou imprevisível. Elas são representadas por letras maiúsculas (X, Y, Z). A função de probabilidade descreve a probabilidade de uma variável aleatória assumir um valor específico. Para variáveis aleatórias discretas (que assumem valores distintos), a função de probabilidade é definida como P(X = x), onde x é o valor específico.

**Parágrafo 2:**

Para variáveis aleatórias contínuas (que podem assumir qualquer valor em um intervalo), a função de densidade de probabilidade é usada em vez da função de probabilidade. Ela é definida como f(x), onde x é o valor da variável aleatória. A área sob a função de densidade de probabilidade representa a probabilidade da variável aleatória cair em um determinado intervalo de valores.

**Parágrafo 3:**

As funções de probabilidade fornecem uma representação matemática das probabilidades associadas à variável aleatória. Elas são essenciais para analisar dados, prever resultados e tomar decisões sob incerteza. A compreensão das variáveis aleatórias e funções de probabilidade é fundamental em estatística, probabilidade e vários outros campos.

Item do edital: Principais distribuições de probabilidade discretas e contínuas: distribuição uniforme

**Distribuições Discretas**

As distribuições de probabilidade discretas modelam o número de ocorrências de um evento em um espaço amostral finito ou contável. As distribuições discretas comuns incluem:

* **Distribuição binomial:** Modela o número de sucessos em um número fixo de ensaios independentes, cada um com uma probabilidade constante de sucesso. Probabilidade de exatamente x sucessos: P(X = x) = (n!/(x!(n-x)!)) \* p^x \* (1-p)^(n-x)
* **Distribuição de Poisson:** Modela o número de ocorrências de um evento em um intervalo fixo de tempo ou espaço, quando a média é conhecida. Probabilidade de exatamente x ocorrências: P(X = x) = (e^(-λ) \* λ^x) / x!

**Distribuições Contínuas**

As distribuições de probabilidade contínuas modelam medições contínuas em um intervalo ou todo o eixo real. As distribuições contínuas comuns incluem:

* **Distribuição normal:** Também conhecida como distribuição gaussiana, é uma distribuição simétrica em forma de sino. Probabilidade de x dentro do intervalo [a, b]: P(a < X < b) = (1/(σ \* √(2π))) \* ∫[a,b] (e^(-(x-μ)^2/(2σ^2))) dx
* **Distribuição uniforme:** Modela situações em que todas as possibilidades são igualmente prováveis dentro de um intervalo especificado. Densidade de probabilidade: f(x) = 1/(b-a) para a < x < b

Item do edital: Distribuições de probabilidade discretas e contínuas: distribuição binomial

**Distribuições de Probabilidade Discretas**

As distribuições de probabilidade discretas descrevem a probabilidade de eventos com um número finito ou contável de resultados. Elas são usadas para situações onde o número de resultados possíveis é limitado, como o número de sucessos em uma série de ensaios ou o número de defeitos em um produto. Uma distribuição binomial é um exemplo clássico de distribuição discreta.

**Distribuição Binomial**

A distribuição binomial é usada para modelar o número de sucessos em um número fixo de ensaios independentes, onde a probabilidade de sucesso é constante para cada ensaio. Seja X uma variável aleatória binominal, então sua probabilidade de massa é dada por:

P(X = k) = (n escolher k) \* p^k \* (1-p)^(n-k)

onde:

* n é o número de ensaios
* k é o número de sucessos
* p é a probabilidade de sucesso em cada ensaio

**Exemplos de Aplicações**

A distribuição binomial é amplamente utilizada em diversas áreas, incluindo:

* Testes de hipóteses para determinar se uma moeda é justa
* Análise de dados de pesquisas para estimar a proporção de uma população que possui uma determinada característica
* Modelagem do número de clientes que visitam uma loja por dia

Item do edital: Distribuições de probabilidade discretas e contínuas: distribuição Poisson

**Distribuições de Probabilidade Discretas**

As distribuições de probabilidade discretas descrevem eventos que assumem um número finito ou contável de valores. A distribuição de Poisson é uma distribuição discreta que modela o número de ocorrências de um evento durante um período de tempo ou espaço fixo.

**Fórmula da Distribuição de Poisson**

A função de probabilidade da distribuição de Poisson é dada por:

P(X = k) = (e^(-λ) \* λ^k) / k!

Onde:

* X é a variável aleatória representando o número de ocorrências
* k é o número específico de ocorrências
* λ é a média da distribuição de Poisson

**Características da Distribuição de Poisson**

A distribuição de Poisson é caracterizada por uma forma assimétrica em J, com a moda sendo igual à média (λ). A variância da distribuição também é igual à média, o que indica que a média é um bom indicador da variabilidade dos dados. A distribuição de Poisson é amplamente utilizada em uma variedade de campos, incluindo bioestatística, controle de qualidade e teoria de filas.

Item do edital: Distribuições de probabilidade discretas e contínuas: distribuição normal

**Distribuições de Probabilidade Discretas e Contínuas**

As distribuições de probabilidade são funções matemáticas que descrevem a probabilidade de ocorrência de diferentes valores em um espaço amostral. Existem dois tipos principais de distribuições de probabilidade: discretas e contínuas.

As distribuições de probabilidade discretas lidam com variáveis ​​que podem assumir apenas um número finito ou enumerável de valores distintos. Exemplos comuns incluem distribuições binomial, poissonian e geométrica. Para uma distribuição de probabilidade discreta X, a função de probabilidade P(X = x) representa a probabilidade de X assumir o valor específico x.

As distribuições de probabilidade contínuas, por outro lado, lidam com variáveis ​​que podem assumir qualquer valor dentro de um intervalo específico. Exemplos notáveis ​​incluem distribuições normal, lognormal e exponencial. Para uma distribuição de probabilidade contínua X com densidade de probabilidade f(x), a probabilidade de X cair em um intervalo específico [a, b] é dada por P(a ≤ X ≤ b) = ∫[a,b] f(x) dx.

Item do edital: Medidas de tendência central

**Medidas de Tendência Central: Média, Mediana e Moda**

As medidas de tendência central fornecem um valor numérico que representa o ponto central ou típico de um conjunto de dados. A **média**, também conhecida como média aritmética, é a soma de todos os valores dividida pelo número de valores no conjunto. Sua fórmula é:

Média = (x1 + x2 + ... + xn) / n

onde x1, x2, ..., xn são os valores do conjunto de dados e n é o número de valores.

A **mediana** é o valor do meio em um conjunto de dados ordenado do menor para o maior. Se houver um número par de valores, a mediana é a média dos dois valores do meio.

A **moda** é o valor que ocorre com mais frequência em um conjunto de dados. Pode haver mais de uma moda em um conjunto de dados.

Item do edital: Medidas de dispersão

**Parágrafo 1:**

As Medidas de Dispersão são estatísticas que medem a variabilidade dos pontos dedados em um conjunto dedados. Elas fornecem uma compreensão da distribuição dosdados, indicando o quão espalhados ou agrupados os pontos dedados estão em torno dovalor médio. As principais Medidas de Dispersão incluem Variânci e Desvio Padrã, que medem a dispersão em torno da média.

**Parágrafo 2:**

A **Variânci** é a média dos quadados das diferenças entre os pontos dedo e a média, e é representada por σ². Quanto maior a variânci, mais dispersosestão os pontos dedados em torno da média. O **Desvio Padrã** é a raiz quadrada davariânci, e é representado por σ. Ele fornece uma medida da dispersão em unidadedas do próprio conjunto dedados, facilitando a comparação da dispersão entre diferentesconjuntos dedados.

**Parágrafo 3:**

Outras Medidas de Dispersão incluem o **Intervalo Interquartil**, que mede a faixa dos50% médios dos pontos dedados, e o **Desvio Interquartil**, que é a metade doIntervalo Interquartil. Elas fornecem uma medida da dispersão que é resistente aoutliers (valores extremos). A **Amplitude**, que é a diferança entre o valor máxima eo mínimo, é uma medida da dispersão geral, mas pode ser influenciada por outliers.

Item do edital: Medidas de correlação.

**Medidas de Correlação**

As medidas de correlação são estatísticas que medem o grau de associação entre duas variáveis. Elas quantificam a direção e a força do relacionamento entre variáveis, indicando se elas variam juntas ou em direções opostas. As medidas de correlação mais comuns incluem a correlação de Pearson (r) e a correlação de Spearman (ρ).

A correlação de Pearson (r) é um coeficiente de correlação que mede o relacionamento linear entre duas variáveis contínuas. Sua fórmula é:

r = Σ(x - x̄)(y - ȳ) / √Σ(x - x̄)²(y - ȳ)²

Onde:

* x e y são os valores das variáveis
* x̄ e ȳ são as médias das respectivas variáveis
* Σ representa a soma

A correlação de Spearman (ρ) é um coeficiente de correlação que mede o relacionamento monotonicamente crescente ou decrescente entre duas variáveis ordinais ou contínuas. Sua fórmula é:

ρ = 1 - 6Σd² / n(n² - 1)

Onde:

* d é a diferença de classificação
* n é o número de pares de dados

Valores de r ou ρ próximos de +1 ou -1 indicam uma correlação forte e positiva ou negativa, respectivamente. Valores próximos de 0 indicam uma correlação fraca ou ausente.

Item do edital: Teorema do limite central.

**Parágrafo 1:**

O Teorema do Limite Central (TLC) afirma que a distribuição amostral das médias de uma amostra aleatória grande (n ≥ 30) de uma população com média μ e desvio padrão σ tenderá para uma distribuição normal, independentemente da distribuição da população original. Isso é expresso pela seguinte fórmula:

X̄ ~ N(μ, σ/√n)

onde:

* X̄ é a média da amostra
* μ é a média da população
* σ é o desvio padrão da população
* n é o tamanho da amostra

**Parágrafo 2:**

O TLC é essencial para inferência estatística, pois permite que façamos inferências sobre a população com base em amostras. Ele afirma que, mesmo que a população tenha uma distribuição assimétrica ou enviesada, a distribuição amostral das médias será aproximadamente normal. Isso significa que podemos usar a distribuição normal para fazer estimativas e testes de hipóteses sobre a população.

**Parágrafo 3:**

A precisão da distribuição normal aumenta com o tamanho da amostra. Quanto maior o tamanho da amostra, menor a variância da distribuição amostral das médias, o que resulta em uma aproximação mais precisa da distribuição da população. O TLC é criticamente importante em pesquisas de opinião, onde as amostras são usadas para fazer inferências sobre grandes populações.

Item do edital: Regra empírica (regra de três sigma) da distribuição normal.

**Parágrafo 1:**

A Regra Empírica, também conhecida como Regra de Três Sigma, é um princípio fundamental da distribuição normal. Ela estabelece que, para qualquer distribuição normal:

* Aproximadamente 68% dos dados estão dentro de um desvio padrão da média (μ).

**Parágrafo 2:**

Ampliando o escopo, a regra afirma que:

* Aproximadamente 95% dos dados estão dentro de dois desvios padrão da média (μ ± 2σ).
* Aproximadamente 99,7% dos dados estão dentro de três desvios padrão da média (μ ± 3σ).

**Parágrafo 3:**

Essas porcentagens podem ser expressas matematicamente usando a curva de Gauss:

P(μ - σ < X < μ + σ) = 0,6827

P(μ - 2σ < X < μ + 2σ) = 0,9545

P(μ - 3σ < X < μ + 3σ) = 0,9973

Observe que a Regra Empírica fornece uma aproximação razoável para a distribuição normal, mas não é exata. Para distribuições não normais, essas porcentagens podem variar.

Item do edital: Diagramas causais: grafos acíclicos dirigidos

**Parágrafo 1:**

Os diagramas causais são representações gráficas de relações causais entre variáveis. Eles são grafos acíclicos dirigidos (DAGs), o que significa que não contêm ciclos e as arestas são orientadas. Cada nó no diagrama representa uma variável, enquanto as arestas representam relacionamentos causais entre elas. As arestas são direcionadas da causa para o efeito.

**Parágrafo 2:**

A estrutura do DAG determina a direção da causalidade e ajuda a identificar as variáveis causais e de efeito. A regra "não há caminho de volta" é essencial para garantir que as relações causais sejam válidas. Isso significa que uma variável não pode ser tanto causa quanto efeito de outra variável no mesmo caminho.

**Parágrafo 3:**

Os diagramas causais são ferramentas importantes para analisar e testar hipóteses sobre relações causais. Eles podem ser usados para calcular efeitos causais usando fórmulas como a regra do produto e a regra da soma. Por exemplo, para calcular o efeito causal de uma variável X sobre uma variável Y, podemos usar a fórmula P(Y|X) = P(Y|X=x1)P(X=x1) + P(Y|X=x2)P(X=x2), onde x1 e x2 são os possíveis valores de X.

Item do edital: Diagramas causais: variáveis confundidoras

**Parágrafo 1:**

Diagramas causais são ferramentas visuais que representam relacionamentos causais entre variáveis. Eles são usados para identificar e analisar fatores que podem influenciar um determinado resultado. No entanto, as variáveis confundidoras podem complicar a interpretação desses diagramas, pois são variáveis que influenciam tanto a causa quanto o efeito.

**Parágrafo 2:**

Uma variável confundidora (C) pode ser representada no diagrama causal como um nó que se conecta à causa (X) e ao efeito (Y). A presença de uma variável confundidora pode distorcer o relacionamento entre X e Y, fazendo com que pareça mais forte ou mais fraco do que realmente é. Isso ocorre porque a variável confundidora pode explicar parte da variação em Y que é atribuída a X.

**Parágrafo 3:**

Um exemplo de uma variável confundidora é a idade em um estudo que investiga a relação entre tabagismo e câncer de pulmão. A idade pode influenciar tanto o tabagismo (pessoas mais velhas são mais propensas a fumar) quanto o câncer de pulmão (pessoas mais velhas são mais propensas a desenvolver câncer de pulmão). Portanto, a idade é uma variável confundidora potencial neste estudo e precisa ser ajustada na análise para evitar distorcer os resultados. Uma fórmula estatística frequentemente usada para controlar variáveis confundidoras é o ajuste por regressão múltipla:

Y = β₀ + β₁X + β₂C + ε

onde:

* Y é a variável de resultado
* X é a variável de exposição
* C é a variável confundidora
* β₀ é a interceptação
* β₁ e β₂ são os coeficientes de regressão
* ε é o termo de erro

Item do edital: Diagramas causais: variáveis colisoras

**Parágrafo 1:**

Em diagramas causais, variáveis colisooras são variáveis que influenciam tanto a variável dependente quanto a variável independente. Isso pode levar à inferência errônea de causalidade entre a variável independente e a variável dependente. Por exemplo, se a variável X influencia tanto Y quanto Z, e Z também influencia Y, então observar uma correlação entre X e Y não prova que X causa Y.

**Parágrafo 2:**

Para identificar variáveis colisooras, é importante considerar todos os possíveis caminhos causais entre a variável independente, a variável dependente e outras variáveis relacionadas. Uma maneira de fazer isso é usar o critério de back-door, que afirma que uma variável X é uma variável colisoora se o seguinte for verdadeiro: 1) X influencia a variável dependente Y; 2) X influencia a variável independente Z; e 3) não há caminho causal não bloqueado de Z para Y, exceto através de X.

**Parágrafo 3:**

A presença de variáveis colisooras pode ser controlada ajustando-se para essas variáveis em modelos estatísticos ou usando métodos como randomização ou correspondência de pontuação de propensão. No entanto, é importante observar que ajustar para variáveis colisooras só é possível se essas variáveis forem observadas ou medidas. Se variáveis colisooras não observadas estiverem presentes, pode ser difícil ou impossível fazer inferências causais válidas.

Item do edital: Diagramas causais: variáveis de mediação.

**Parágrafo 1:**

Os diagramas causais são representações gráficas que mostram as relações causais entre variáveis em um sistema. Uma variável de mediação é uma variável que transmite o efeito de uma variável independente sobre uma variável dependente. Em outras palavras, a variável de mediação é influenciada pela variável independente e, por sua vez, influencia a variável dependente.

**Parágrafo 2:**

O efeito de mediação é quantificado pelo efeito indireto, que é o produto dos efeitos da variável independente na variável mediadora e da variável mediadora na variável dependente. O efeito direto é o efeito da variável independente na variável dependente na ausência da variável mediadora. O efeito total é a soma do efeito direto e do efeito indireto.

**Parágrafo 3:**

Matematicamente, o efeito de mediação pode ser representado como:

M = c \* b

I = a \* b

T = a \* c + b \* c

onde:

* M é o efeito de mediação (efeito indireto)
* I é o efeito direto
* T é o efeito total
* a é o efeito da variável independente na variável mediadora
* b é o efeito da variável mediadora na variável dependente
* c é o efeito da variável independente na variável dependente na ausência da variável mediadora

Item do edital: Métodos e técnicas de identificação causal: Métodos experimentais RCT

**Parágrafo 1:**

Os métodos experimentais, particularmente os ensaios clínicos randomizados (RCTs), são considerados o padrão ouro para identificação causal devido à sua capacidade de eliminar vieses de seleção e confusão. Em um RCT, os participantes são aleatoriamente atribuídos a um grupo de tratamento ou controle. Essa randomização garante que os grupos sejam balanceados em relação a todos os fatores conhecidos e desconhecidos que podem influenciar o desfecho.

**Parágrafo 2:**

A diferença nos desfechos entre os grupos de tratamento e controle pode ser atribuída ao tratamento em estudo. Isso ocorre porque a randomização cria grupos comparáveis e elimina quaisquer diferenças sistemáticas entre eles. O efeito causal do tratamento pode ser estimado calculando o risco relativo (RR) ou a diferença de risco (DR) entre os grupos:

**RR = Risco no grupo de tratamento / Risco no grupo de controle**

**DR = Risco no grupo de tratamento - Risco no grupo de controle**

**Parágrafo 3:**

Os RCTs oferecem evidências fortes de causalidade, mas também podem ter limitações, como custo, viabilidade e generalização para populações mais amplas. Além disso, pode ser antiético conduzir RCTs para certas intervenções, como tratamentos que são conhecidos por serem prejudiciais.

Item do edital: Métodos e técnicas de identificação causal: métodos de identificação quase-experimental.

**Métodos Quase-Experimentais**

Os métodos quase-experimentais são técnicas de identificação causal que visam aproximar-se de experimentos controlados em situações em que a randomização não é possível. Eles envolvem a comparação de grupos tratados e de controle que não foram randomizados, mas que possuem características semelhantes. Um método comum é a regressão em descontinuidades regressivas (RDD), que aproveita as descontinuidades no tratamento com base em um valor de corte (por exemplo, pontuação em um teste) para estimar o efeito causal do tratamento.

**Fórmula para RDD:**

Y = β0 + β1 \* (T > c) + β2 \* X + ε

Onde:

* Y é a variável de resposta
* T é a variável de tratamento (1 se T > c, 0 caso contrário)
* c é o valor de corte
* X são as covariáveis de controle
* ε é o termo de erro

**Outros Métodos Quase-Experimentais:**

Outros métodos quase-experimentais incluem:

* **Escore de Propensão Ponderada:** Atribui pesos diferentes às observações tratadas e de controle para equilibrar os grupos com base nas covariáveis observadas.
* **Modelos de Diferenças em Diferenças:** Comparação das diferenças entre os grupos tratados e de controle antes e depois do tratamento.
* **Experimentos Naturais:** Exploração de eventos naturais (por exemplo, desastres ou mudanças políticas) que criam condições semelhantes a experimentos.

Item do edital: Tipos de viés no processo gerador dos dados e soluções: Sampling bias

**Sampling Bias (Viés de Amostragem)**

Sampling bias ocorre quando o processo de amostragem não seleciona aleatoriamente um subconjunto representativo da população, levando a uma distorção nos dados coletados. Por exemplo, se um pesquisador conduz uma pesquisa online e a maioria dos participantes são pessoas que têm acesso à internet, os resultados da pesquisa podem superestimar as opiniões de pessoas com acesso à internet em detrimento daquelas sem acesso.

**Fórmula:**

O viés de amostragem pode ser calculado como:

Viés de amostragem = (Proporção na amostra - Proporção na população) x 100%

**Soluções:**

Existem várias soluções para minimizar o viés de amostragem, incluindo:

* **Amostragem aleatória simples:** Selecionar um subconjunto aleatório da população sem substituição.
* **Amostragem estratificada:** Dividir a população em estratos com base em características relevantes e selecionar aleatoriamente de cada estrato.
* **Amostragem por conglomerados:** Dividir a população em conglomerados e selecionar aleatoriamente um subconjunto de conglomerados, incluindo todos os indivíduos dentro dos conglomerados selecionados.

Item do edital: Tipos de viés no processo gerador dos dados e soluções: Selection bias

**Selection bias** ocorre quando os indivíduos incluídos em um estudo são selecionados de forma não aleatória, levando a uma amostra tendenciosa que não representa a população inteira. Por exemplo, se um pesquisador coleta dados de um grupo de voluntários, os resultados podem estar distorcidos para aqueles que estão mais motivados ou têm mais tempo para participar.

Uma solução para o selection bias é usar técnicas de amostragem aleatória, como amostragem aleatória simples, amostragem aleatória estratificada ou amostragem por conglomerados. Essas técnicas garantem que cada indivíduo tenha uma chance igual de ser selecionado, reduzindo o risco de uma amostra tendenciosa.

Além disso, os pesquisadores podem usar técnicas de ponderação para ajustar os dados de uma amostra tendenciosa. A ponderação envolve atribuir pesos diferentes aos indivíduos com base em suas probabilidades de seleção. Isso ajuda a compensar o viés de seleção e produzir uma amostra mais representativa.

Item do edital: Tipos de viés no processo gerador dos dados e soluções: Attrition bias

**Viés de Atrição**

O viés de atrição ocorre quando há uma perda significativa de participantes em um estudo ou experimento, levando a uma amostra não representativa. Isso pode resultar em estimativas tendenciosas e conclusões enganosas.

Um exemplo clássico de viés de atrição é o "viés do trabalhador selecionado", que ocorre quando os participantes que deixam um estudo (por exemplo, abandonam o emprego) são sistematicamente diferentes daqueles que permanecem. Isso pode levar a uma superestimação do efeito de uma intervenção ou tratamento, pois os participantes que desistiram podem ter sido menos propensos a responder positivamente.

Para minimizar o viés de atrição, é essencial tomar medidas para manter as taxas de atrito baixas. Isso pode envolver o uso de estratégias de recrutamento e retenção eficazes, como oferecer incentivos por participação e acompanhamento regular dos participantes. Quando ocorre a atrito, é crucial avaliar se ela é representativa da população do estudo. Se não for, métodos estatísticos como análise de intenção de tratar ou ponderação inversa da probabilidade de abandono podem ser usados para ajustar os resultados.

Item do edital: Tipos de viés no processo gerador dos dados e soluções: Reporting bias

**Viés de Relato**

O viés de relato refere-se à tendência dos indivíduos de relatar informações de forma imprecisa ou incompleta. Pode ocorrer devido a fatores como memória falha, pressão social ou desejo de apresentar-se favoravelmente. Por exemplo, os pacientes podem subestimar o consumo de tabaco para evitar julgamentos ou embaraço.

**Implicações**

O viés de relato pode prejudicar a precisão dos dados coletados, levando a estimativas enviesadas e conclusões erradas. Em estudos epidemiológicos, pode subestimar a prevalência de certos comportamentos ou condições, distorcendo os resultados da pesquisa.

**Soluções**

Existem várias estratégias para mitigar o viés de relato:

* **Uso de métodos anônimos de coleta de dados:** Isso reduz a pressão social e incentiva respostas honestas.
* **Treinamento de entrevistadores:** Treinamento adequado pode melhorar a capacidade dos entrevistadores de obter informações precisas e evitar influenciar as respostas.
* **Validação das respostas:** A comparação das respostas com outras fontes de informações (por exemplo, registros médicos) pode ajudar a verificar a precisão.

Item do edital: Tipos de viés no processo gerador dos dados e soluções: Measurement bias.

**Viés de Medição**

O viés de medição ocorre quando o processo de coleta de dados introduz erros ou imprecisões que distorcem os resultados. Isso pode acontecer devido a instrumentos de medição defeituosos, métodos de coleta de dados inadequados ou entrevistadores tendenciosos. Por exemplo, uma balança que mede consistentemente o peso das pessoas 1 kg mais leve do que o real introduzirá um viés de medição nos dados de peso coletados.

**Solução:**

Para resolver o viés de medição, é essencial calibrar e manter regularmente os instrumentos de medição para garantir sua precisão. Métodos de coleta de dados devem ser cuidadosamente projetados para minimizar erros e controlar variáveis ​​estranhas. Além disso, os entrevistadores devem ser treinados para conduzir entrevistas imparciais e minimizar vieses pessoais.

**Fórmula:**

O viés de medição pode ser quantificado usando uma fórmula:

Viés de Medição = Valor Medido - Valor Real

Esta fórmula calcula a diferença entre o valor medido e o valor verdadeiro conhecido, fornecendo uma medida do erro de medição.

Item do edital: Modelos probabilísticos gráficos: cadeias de Markov

**Parágrafo 1:**

Modelos probabilísticos gráficos (GPMs) são uma ferramenta poderosa para representar e raciocinar sobre probabilidades complexas. Entre os GPMs, as **cadeias de Markov** são comumente usadas para modelar sistemas sequenciais onde o estado atual depende apenas do estado anterior. Cada estado é representado por um nó no grafo GPM, enquanto as arestas representam as probabilidades de transição entre os estados.

**Parágrafo 2:**

Formalmente, uma cadeia de Markov de ordem *n* é definida pela distribuição de probabilidade conjunta:

P(X\_1, X\_2, ..., X\_t) = P(X\_1) \* P(X\_2 | X\_1) \* ... \* P(X\_t | X\_{t-1})

onde *X\_i* são as variáveis de estado nas posições *i*. A probabilidade de transição condicional de um estado *X\_t* para o próximo estado *X\_{t+1}* é dada por:

P(X\_{t+1} | X\_t)

**Parágrafo 3:**

As cadeias de Markov são amplamente utilizadas em aprendizado de máquina, processamento de linguagem natural e bioinformática. Elas podem ser usadas para modelar sequências de dados, prever eventos futuros e inferir estados ocultos. Por exemplo, uma cadeia de Markov poderia ser usada para modelar o movimento diário do preço das ações, onde o preço atual depende apenas do preço do dia anterior.

Item do edital: Modelos probabilísticos gráficos: filtros de Kalman

**Parágrafo 1:**

Os Modelos Probabilísticos Gráficos (MPG) são estruturas matemáticas que representam a dependência conjunta entre variáveis. Os filtros de Kalman são um tipo específico de MPG que estima o estado oculto de um sistema dinâmico observando as suas saídas. Eles fornecem uma distribuição de probabilidade para o estado do sistema, dada uma sequência de observações. A equação de atualização do filtro de Kalman é dada por:

x\_t = A \* x\_{t-1} + B \* u\_t + K \* (z\_t - H \* x\_{t-1})

onde x\_t é o estado estimado no tempo t, A é a matriz de transição do estado, B é a matriz de entrada, u\_t é o vetor de entrada, K é o ganho de Kalman, z\_t é a observação no tempo t, e H é a matriz de observação.

**Parágrafo 2:**

A vantagem dos filtros de Kalman é que eles levam em conta a incerteza nas observações e no estado do sistema. Eles atualizam a distribuição de probabilidade do estado à medida que novas observações são recebidas, tornando-os adaptáveis a sistemas dinâmicos em mudança. A equação de predição do filtro de Kalman é dada por:

x\_{t+1} = A \* x\_t + B \* u\_t

**Parágrafo 3:**

Os filtros de Kalman são amplamente utilizados em aplicações como navegação, controle de processos e processamento de sinais. Eles são particularmente valiosos em situações em que o estado do sistema não é diretamente observável ou é ruidoso. Ao fornecer uma estimativa confiável do estado oculto, os filtros de Kalman permitem decisões mais informadas e melhor desempenho do sistema.

Item do edital: Modelos probabilísticos gráficos: Redes bayesianas.

**Parágrafo 1:**

As redes bayesianas são modelos gráficos probabilísticos que representam a dependência entre variáveis aleatórias usando um grafo direcionado. Cada nó no grafo representa uma variável aleatória, e os arcos direcionados indicam as relações causais entre as variáveis. A distribuição conjunta de todas as variáveis é fatorada de acordo com a estrutura do grafo:

P(X₁, X₂, ..., Xₙ) = ∏ᵢ P(Xᵢ | Pa(Xᵢ))

onde Pa(Xᵢ) são os pais de Xᵢ no grafo.

**Parágrafo 2:**

As redes bayesianas permitem o cálculo de distribuições condicionais e probabilidades posteriores de forma eficiente usando propagação de crenças. Além disso, elas podem ser aprendidas a partir de dados observados usando métodos como estimativa de máxima verossimilhança ou algoritmos de otimização. As redes bayesianas são amplamente utilizadas em uma variedade de aplicações, incluindo raciocínio probabilístico, aprendizado de máquina e processamento de linguagem natural.

**Parágrafo 3:**

As redes bayesianas são uma ferramenta poderosa para modelar a incerteza e o conhecimento em domínios complexos. No entanto, elas podem se tornar computacionalmente caras para grafos grandes ou quando a distribuição envolve muitas variáveis. Além disso, elas assumem independência condicional entre as variáveis, o que pode não sempre ser realista. Apesar dessas limitações, as redes bayesianas continuam sendo uma técnica valiosa para representar e raciocinar com incerteza em vários campos.

Item do edital: Testes de hipóteses: teste-z

**Parágrafo 1:** Os testes de hipóteses são métodos estatísticos usados para decidir se uma hipótese sobre uma população é apoiada pelos dados observados. O teste-z é um teste específico usado quando a média da população é desconhecida, mas a distribuição da amostra é aproximadamente normal.

**Parágrafo 2:** Para realizar um teste-z, são calculados o escore-z, que mede o desvio da média da amostra em relação à média hipotética, e o valor-p, que mede a probabilidade de obter um escore-z tão extremo ou mais extremo, assumindo que a hipótese nula seja verdadeira.

**Parágrafo 3:** Se o valor-p for menor do que um nível de significância predeterminado (geralmente 0,05), a hipótese nula é rejeitada e conclui-se que há evidências para apoiar a hipótese alternativa. Caso contrário, a hipótese nula não é rejeitada e conclui-se que não há evidências suficientes para apoiar a hipótese alternativa. A fórmula para o escore-z é:

z = (x̄ - μ) / (σ / √n)

onde x̄ é a média da amostra, μ é a média hipotética, σ é o desvio padrão da população (se conhecido) ou da amostra (se desconhecido) e n é o tamanho da amostra.

Item do edital: Testes de hipóteses: teste-t

**Teste-t**

O teste-t é um teste estatístico usado para determinar se a média de uma população difere de um valor específico. Ele pressupõe uma distribuição normal e é usado quando o tamanho da amostra é pequeno (geralmente menos de 30). A seguir está a fórmula para o teste-t:

t = (x̄ - μ) / (s / √n)

onde:

* x̄ é a média da amostra
* μ é a média populacional hipotética
* s é o desvio padrão da amostra
* n é o tamanho da amostra

O valor obtido do teste-t deve ser comparado com o valor crítico do teste-t, que é obtido a partir de uma tabela de distribuição t com n-1 graus de liberdade. Se o valor do teste-t for maior que o valor crítico, então a hipótese nula (de que a média populacional é igual ao valor hipotético) é rejeitada.

**Interpretação**

Um valor de p significativo (geralmente menor que 0,05) indica que a diferença entre a média da amostra e o valor hipotético é estatisticamente significativa. Isso significa que é improvável que a diferença tenha ocorrido por acaso. Por outro lado, um valor de p não significativo indica que a diferença não é estatisticamente significativa e pode ter ocorrido por acaso.

Item do edital: Testes de hipóteses: valor-p

Em testes de hipóteses, o valor-p é uma medida estatística que representa a probabilidade de se obter um resultado pelo menos tão extremo quanto o observado, assumindo que a hipótese nula (H0) seja verdadeira. Ele indica a força das evidências contra H0. Valores menores de p indicam evidências mais fortes contra H0, enquanto valores maiores de p sugerem que os resultados observados são mais consistentes com H0.

Formalmente, o valor-p é calculado como a área sob a curva de distribuição de amostragem da estatística de teste, além do valor observado da estatística. Em testes unilaterais, a área é calculada em uma direção (cauda), enquanto em testes bilaterais a área é calculada em ambas as direções. As fórmulas usadas para calcular o valor-p variam dependendo da estatística de teste e da distribuição de amostragem. Por exemplo, para um teste t de uma amostra:

p = 2 \* P(t < t\_obs)

onde t\_obs é o valor observado do t-score e P(t < t\_obs) é a área sob a curva t-distribuição à esquerda de t\_obs.

O valor-p é essencial para os testes de hipóteses, pois fornece um critério objetivo para aceitar ou rejeitar H0. Normalmente, um nível de significância (α) é definido antes do teste, e se o valor-p for menor que α, H0 é rejeitada. No entanto, é importante observar que o valor-p representa apenas a probabilidade de obter um resultado extremo sob H0 e não prova ou refuta H0.

Item do edital: Testes de hipóteses: testes para uma amostra

**Parágrafo 1:** Os testes de hipóteses são procedimentos estatísticos usados para determinar se uma afirmação sobre uma população (chamada hipótese) é suportada por evidências amostrais. Em testes de uma amostra, comparamos o valor observado de uma estatística de amostra com um valor esperado para determinar se a diferença é estatisticamente significativa. A hipótese nula (H0) afirma que não há diferença entre o valor observado e o valor esperado, enquanto a hipótese alternativa (Ha) afirma que existe uma diferença.

**Parágrafo 2:** Para realizar um teste de hipótese de uma amostra, usamos uma distribuição de probabilidade apropriada para calcular a probabilidade de observar o valor da estatística da amostra ou um valor mais extremo, assumindo que a hipótese nula seja verdadeira. Esta probabilidade é chamada de valor p. Se o valor p for menor que um nível de significância predeterminado (geralmente 0,05), rejeitamos a hipótese nula e concluímos que a diferença é estatisticamente significativa. Caso contrário, não rejeitamos H0.

**Parágrafo 3:** Fórmulas comuns para testar hipóteses incluem:

* **Teste z:** µ = µ0, onde µ é a média da população e µ0 é o valor esperado da média da amostra
* **Teste t:** µ = µ0, onde µ é a média da população e µ0 é o valor esperado da média da amostra, e o desvio padrão da população é desconhecido
* **Teste de qui-quadrado:** x2 = Σ [(O - E)2 / E], onde O é o valor observado, E é o valor esperado e Σ é a soma sobre todas as categorias

Item do edital: Testes de hipóteses: testes de comparação de duas amostras

**Testes de Comparação de Duas Amostras**

Os testes de hipóteses comparam duas amostras independentes para determinar se diferem significativamente em termos de médias ou proporções. Eles testam a hipótese nula de que as médias (ou proporções) populacionais das duas amostras são iguais e calculam o valor p, que representa a probabilidade de obter resultados tão extremos ou mais extremos que os observados, assumindo que a hipótese nula seja verdadeira.

Para testar a diferença nas médias de duas amostras independentes, o teste t de duas amostras é usado. A estatística do teste é:

t = (x1 - x2) / (sqrt((s1^2 / n1) + (s2^2 / n2)))

onde x1 e x2 são as médias das amostras, s1 e s2 são os desvios padrão das amostras e n1 e n2 são os tamanhos das amostras.

Para testar a diferença nas proporções de duas amostras independentes, o teste z para proporções é usado. A estatística do teste é:

z = (p1 - p2) / sqrt(p1 \* (1 - p1) / n1 + p2 \* (1 - p2) / n2)

onde p1 e p2 são as proporções das amostras e n1 e n2 são os tamanhos das amostras.

Item do edital: Testes de hipóteses: teste de normalidade (chi square)

**Parágrafo 1:**

Os testes de hipóteses são usados em estatística para determinar se existe uma diferença significativa entre os dados observados e os esperados. O teste de normalidade é um tipo de teste de hipótese que determina se os dados seguem uma distribuição normal. A distribuição normal, também conhecida como distribuição gaussiana, é uma distribuição simétrica e em forma de sino que é comum em muitos fenômenos naturais.

**Parágrafo 2:**

O teste de normalidade qui-quadrado (χ²) é um método comum para testar a normalidade. Ele compara a frequência observada dos dados em diferentes intervalos com a frequência esperada sob a suposição de uma distribuição normal. O teste estatístico qui-quadrado é calculado como:

χ² = Σ [(O - E)² / E]

Onde:

* O = frequência observada
* E = frequência esperada

**Parágrafo 3:**

O valor qui-quadrado é comparado a um valor crítico para determinar se existe uma diferença significativa entre as frequências observadas e esperadas. Se o valor qui-quadrado for maior que o valor crítico, é rejeitada a hipótese nula de que os dados seguem uma distribuição normal. Um valor-p também pode ser calculado a partir do valor qui-quadrado para fornecer uma medida da significância estatística. Um valor-p baixo indica que há uma diferença significativa entre os dados observados e esperados, e, portanto, os dados não seguem uma distribuição normal.

Item do edital: Testes de hipóteses: intervalos de confiança.

**Paragrafo 1:** Em testes de hipóteses, intervalos de confiança são usados para estimar o intervalo de valores possíveis para um parâmetro desconhecido da população, com um determinado nível de confiança. O intervalo de confiança é calculado usando uma fórmula que incorpora a estatística da amostra, o erro padrão da estatística e o nível de confiança desejado.

**Paragrafo 2:** Por exemplo, para estimar o intervalo de confiança para a média populacional (µ) com um nível de confiança de 95%, a fórmula é:

x̄ ± z \* (σ/√n)

Onde:

* x̄ é a média amostral
* σ é o desvio padrão da população (se conhecido) ou o desvio padrão estimado da amostra
* n é o tamanho da amostra
* z é o valor crítico do escore z para o nível de confiança de 95% (1,96)

**Paragrafo 3:** Se o intervalo de confiança não incluir o valor hipotetizado para o parâmetro, isso indica que a hipótese nula deve ser rejeitada com o nível de confiança especificado. Por outro lado, se o intervalo de confiança incluir o valor hipotetizado, a hipótese nula não pode ser rejeitada.

Item do edital: Histogramas e curvas de frequência

\*\*Histograma \*\*

Um histograma é uma representação visual que mostra a frequência de um dado em um intervalo de valores. Os dados são divididos em intervalos (bins) onde cada intervalo é representado por uma barra. A altura da barra é proporcional à frequência dos dados naquele intervalo. O histograma permite visualizar a distribuição de frequência dos dados e identificar padrões e tendências.

**Curva de frequência**

Uma Curva de Frequência é um gráfico que mostra a relação entre a frequência relativa de um dado e seu valor. A frequência relativa é o número de vezes que um dado valor ocorre dividido pelo número total de dados. A Curva de Frequência é uma representação suave da distribuição de frequência e permite identificar a frequência média, o desvio padrão e a assimetria dos dados.

**Fórmulas**

* **Média:** (Σx) / n
* **Desvio Padrão:** √[(Σ(x - (Σx) / n)²)] / n
* **Assimetria:** (Σ(x³ - (Σx²) / n)³) / n

Item do edital: Diagrama boxplot

Um diagrama boxplot é uma representação gráfica de um conjunto de dados numéricos que fornece informações sobre sua distribuição. Ele divide os dados em quartis (Q1, Q2 e Q3), sendo Q2 a mediana.

O boxplot mostra a distribuição dos dados dentro da faixa interquartil (IQR), que é calculada como IQR = Q3 - Q1. A caixa abrange o IQR e é dividida pela mediana. O bigode se estende até 1,5 IQR de Q1 (bigode inferior) e Q3 (bigode superior). Quaisquer dados fora desta faixa são considerados valores discrepantes e são representados por pontos.

Com base nessas informações, o boxplot pode fornecer insights sobre a tendência central, a dispersão e a presença de valores discrepantes nos dados. Ele é uma ferramenta valiosa para comparar distribuições de dados e identificar outliers, ajudando na interpretação e compreensão de grandes conjuntos de dados.

Item do edital: Avaliação de outliers.

**Avaliação de Outliers**

A avaliação de outliers é crucial para identificar e lidar com dados que se desviam significativamente do restante do conjunto de dados. Outliers podem distorcer os resultados de análise estatística, tornando essencial sua detecção e tratamento adequados. Existem vários métodos para avaliar outliers, incluindo regras baseadas em intervalos, testes estatísticos e métodos de aprendizado de máquina.

Um método comum para detectar outliers é a regra do intervalo interquartil (IQR). Essa regra define um outlier como um valor abaixo de Q1 - 1,5 \* IQR ou acima de Q3 + 1,5 \* IQR, onde Q1 é o primeiro quartil, Q3 é o terceiro quartil e IQR é o intervalo interquartil (Q3 - Q1).

Além das regras baseadas em intervalos, os testes estatísticos também podem ser usados para avaliar outliers. O teste de Grubbs, por exemplo, testa se o valor mais extremo em um conjunto de dados é um outlier. O teste de Grubbs usa a fórmula G = |X - μ| / σ, onde X é o valor extremo, μ é a média e σ é o desvio padrão. Se G for maior que um valor crítico, o valor extremo é considerado um outlier.

Item do edital: Técnicas de classificação: Naive Bayes

**Parágrafo 1:**

Técnicas de classificação são usadas para prever a categoria a que pertence um novo dado. Naive Bayes é um algoritmo de classificação probabilístico que aplica o Teorema de Bayes para calcular a probabilidade de um dado pertencer a uma determinada classe. Ele assume que os recursos do dado são independentes entre si, o que simplifica o cálculo.

**Parágrafo 2:**

A fórmula do Teorema de Bayes, que é usada por Naive Bayes, é dada por:

P(C|X) = (P(X|C) \* P(C)) / P(X)

onde:

* P(C|X) é a probabilidade de pertencer à classe C dado o dado X
* P(X|C) é a probabilidade de observar o dado X dada a classe C
* P(C) é a probabilidade da classe C
* P(X) é a probabilidade do dado X

**Parágrafo 3:**

Para classificar um novo dado, Naive Bayes calcula a probabilidade de cada classe e seleciona a classe com a probabilidade mais alta. Apesar de sua simplicidade e eficiência, Naive Bayes pode não ser preciso quando as suposições de independência dos recursos não são atendidas.

Item do edital: Técnica de classificação Regressão logística

**Parágrafo 1:** A Regressão Logística é uma técnica de classificação estatística que modela a probabilidade de um evento binário ocorrer. É apropriada para dados onde a variável dependente é categórica com apenas dois valores (0 ou 1). A função logística sigmoide é usada para mapear os valores da variável independente (x) para uma probabilidade entre 0 e 1.

**Parágrafo 2:** A equação da função logística sigmoide é:

p = 1 / (1 + e^(-x))

onde p é a probabilidade de o evento ocorrer e x é uma combinação linear de variáveis independentes (x1, x2, ..., xn) multiplicadas por seus respectivos coeficientes (β0, β1, ..., βn).

**Parágrafo 3:** O objetivo da Regressão Logística é encontrar os coeficientes β que melhor se ajustam aos dados observados. Isso é feito minimizando uma função de custo, como a função de perda de entropia cruzada. A significância dos coeficientes é testada usando testes estatísticos (por exemplo, razão de verossimilhança), e os coeficientes podem ser usados para calcular a probabilidade de ocorrência do evento para novos dados.

Item do edital: Técnica de classificação Redes neurais artificiais

**Parágrafo 1:**

As redes neurais artificiais (ANNs) são uma classe de algoritmos de aprendizado de máquina inspirados na estrutura e função do cérebro humano. Elas são compostas por camadas de neurônios conectados, onde cada neurônio recebe entradas, processa-as e produz uma saída. As ANNs são capazes de aprender padrões complexos em dados e fazer previsões ou tomar decisões com base nesses padrões.

**Parágrafo 2:**

Para executar a classificação, as ANNs são treinadas em um conjunto de dados rotulado, onde cada dado de entrada é mapeado para uma classe específica. Durante o treinamento, as ponderações e os vieses dos neurônios são ajustados para minimizar uma função de perda, que quantifica a diferença entre as saídas da ANN e as classes de destino. A função de perda comummente usada é a entropia cruzada, definida como:

H(p, q) = - Σ (p(x) log q(x))

onde p representa a probabilidade verdadeira e q representa a probabilidade prevista.

**Parágrafo 3:**

Após o treinamento, a ANN pode classificar novos dados de entrada estimando as probabilidades de cada classe e atribuindo a classe com a maior probabilidade. As ANNs são particularmente adequadas para problemas de classificação em que os dados são não lineares, de alta dimensão ou ruidosos.

Item do edital: Técnica de classificação Árvores de decisão (algoritmos ID3 e C4.5)

**Parágrafo 1:** As árvores de decisão são um algoritmo de aprendizado supervisionado para classificação que constrói uma estrutura de decisão hierárquica. Começa com um nó raiz representando o conjunto de dados bruto. Em cada nó, o algoritmo seleciona o atributo mais informativo para dividir o conjunto de dados em subconjuntos menores com base nos valores do atributo. Este processo é repetido recursivamente até que cada nó represente uma classe ou uma distribuição de classes.

**Parágrafo 2:** O algoritmo ID3 (Iterative Dichotomiser 3) usa a métrica de ganho de informação para selecionar o atributo mais informativo:

Ganho de informação(A) = Entropia(S) - Soma(Valor(A, v) \* Entropia(S(v)))

onde S é o conjunto de dados atual, A é o atributo a ser avaliado, v é cada valor possível de A e S(v) é o subconjunto de S para o qual A tem valor v.

**Parágrafo 3:** O algoritmo C4.5 é uma extensão do ID3 que usa a métrica de ganho de razão para selecionar o atributo mais informativo, que considera o tamanho dos subconjuntos resultantes:

Ganho de razão(A) = (Entropia(S) - Soma(Valor(A, v) \* Entropia(S(v)))) / Entropia(S)

O C4.5 também trata valores ausentes de atributos e poda a árvore resultante para melhorar sua precisão e prevenir o sobreajuste.

Item do edital: Técnica de classificação florestas aleatórias (random forest)

As florestas aleatórias são um método de aprendizado de máquina de conjunto usado para tarefas de classificação e regressão. Elas consistem em um ensemble de árvores de decisão, onde cada árvore é construída usando um subconjunto aleatório dos dados de treinamento. A previsão final é feita por meio do voto majoritário das previsões das árvores individuais.

Formalmente, seja X = {x\_1, ..., x\_n} o conjunto de dados de treinamento, Y = {y\_1,..., y\_n} as variáveis de saída de destino e K o número de árvores no conjunto. Para cada árvore de decisão t ∈ {1, ..., K}, as seguintes etapas são executadas:

* Um subconjunto de treinamento é amostrado aleatoriamente com substituição de X.
* Uma árvore de decisão é construída usando o subconjunto de treinamento amostrado.
* O erro de predição fora da amostra para a árvore t é calculado usando um conjunto de validação.

A previsão final da floresta aleatória é feita pela média das previsões das árvores individuais:

\hat{y} = mean(\hat{y}\_1, \hat{y}\_2, ..., \hat{y}\_K)

onde \hat{y}\_t é a previsão da árvore t.

Item do edital: Técnica de classificação Máquinas de vetores de suporte (SVM – support vector machines)

**Parágrafo 1:** As Máquinas de Vetores de Suporte (SVM) são um poderoso algoritmo de aprendizado de máquina supervisionado usado para classificação e regressão. Elas buscam uma função de decisão que separa dados de classes diferentes com a maior margem possível. A margem é a distância entre a função de decisão e os pontos de dados mais próximos de cada classe, chamados vetores de suporte.

**Parágrafo 2:** Para dados linearmente separáveis, a função de decisão do SVM é uma linha ou hiperplano que maximiza a margem. Para dados não linearmente separáveis, as SVM usam um kernel para mapear os dados para um espaço de dimensão superior, onde eles podem se tornar linearmente separáveis. O kernel mais comum é o Gaussiano:

K(x, y) = exp(-||x - y||^2 / 2σ^2)

onde σ é um parâmetro de largura de banda.

**Parágrafo 3:** As SVM podem ser usadas para vários tipos de problemas de classificação, incluindo problemas binários e multiclasse. Elas são conhecidas por sua alta precisão e eficiência, mesmo com dados de alta dimensão. No entanto, elas podem ser sensíveis a outliers e exigem um ajuste cuidadoso de parâmetros, como o kernel e o parâmetro de regularização.

Item do edital: Técnica de classificação K vizinhos mais próximos (KNN – K-nearest neighbours).

A Classificação K Vizinhos Mais Próximos (KNN) é um algoritmo de classificação supervisionado que atribui uma classe a um novo ponto de dados com base na maioria das classes de seus k vizinhos mais próximos no espaço de recurso.

O algoritmo KNN funciona seguindo estas etapas:

1. Calcule a distância entre o novo ponto de dados e cada ponto de dados no conjunto de treinamento.
2. Classifique os pontos de dados no conjunto de treinamento pelas distâncias calculadas, do menor ao maior.
3. Selecione os k vizinhos mais próximos do novo ponto de dados.
4. Atribua ao novo ponto de dados a classe majoritária entre os k vizinhos mais próximos.

Matematicamente, a distância entre dois pontos de dados x e y no espaço de recurso é normalmente calculada usando a distância euclidiana:

dist(x, y) = sqrt((x1 - y1)^2 + (x2 - y2)^2 + ... + (xn - yn)^2)

onde x1, x2, ..., xn e y1, y2, ..., yn são os valores dos recursos para os pontos de dados x e y, respectivamente.

Item do edital: Técnica de classificação

**Parágrafo 1:**

A classificação é uma técnica estatística supervisionada que atribui rótulos a instâncias desconhecidas baseando-se em um conjunto de instâncias marcadas conhecidas. O objetivo é construir um modelo que mapeie as características de entrada para rótulos de saída. Existem vários algoritmos de classificação, incluindo regressão logística, máquinas de vetores de suporte e árvores de decisão.

**Parágrafo 2:**

A regressão logística é um algoritmo de classificação linear que usa uma função sigmoide para prever a probabilidade de uma instância pertencer a uma classe específica. A função sigmoide é definida como:

f(x) = 1 / (1 + e^(-x))

Onde x é uma combinação linear das características de entrada.

**Parágrafo 3:**

As máquinas de vetores de suporte são um algoritmo de classificação não linear que cria um hiperplano para separar instâncias de classes diferentes. O hiperplano é definido como:

w^T x + b = 0

Onde w é o vetor de pesos, x é o vetor de características de entrada e b é o viés. O objetivo é maximizar a margem entre o hiperplano e as instâncias mais próximas, chamadas de vetores de suporte.

Item do edital: Avaliação de modelos de classificação: treinamento

**Parágrafo 1:**

A avaliação de modelos de classificação é crucial para medir sua eficácia. O objetivo é avaliar a capacidade do modelo de prever corretamente a classe alvo. Para isso, os dados são divididos em conjuntos de treinamento e teste. O modelo é treinado no conjunto de treinamento e sua performance é avaliada no conjunto de teste, que não foi visto durante o treinamento.

**Parágrafo 2:**

Existem várias métricas de avaliação comumente usadas, como precisão, recall e pontuação F1. A precisão mede a proporção de previsões corretas, enquanto o recall mede a proporção de instâncias positivas corretamente classificadas. A pontuação F1 é uma média harmônica de precisão e recall. Dependendo do problema específico, outras métricas, como curva ROC ou lucro da curva de elevação, podem ser mais adequadas.

**Parágrafo 3:**

A avaliação de modelos de classificação também envolve o ajuste de hiperparâmetros, como taxa de aprendizado e tamanho do lote. Esses hiperparâmetros afetam o desempenho do modelo e podem ser ajustados por meio de métodos como validação cruzada ou pesquisa de grade. O objetivo é encontrar o conjunto de hiperparâmetros que resulta no melhor desempenho no conjunto de avaliação.

Item do edital: Avaliação de modelos de classificação: teste

**Parágrafo 1:** A avaliação de modelos de classificação envolve avaliar o desempenho do modelo em dados não vistos. Isso é crucial para determinar a eficácia do modelo em generalizar para novas observações. Métricas comuns incluem precisão, recall, pontuação F1 e curva ROC-AUC. A precisão mede a proporção de classificações corretas em todas as classificações, enquanto o recall mede a proporção de exemplos positivos classificados corretamente. A pontuação F1 é um equilíbrio entre precisão e recall, e a curva ROC-AUC mede a capacidade do modelo de distinguir entre classes positivas e negativas.

**Parágrafo 2:** Para avaliar a robustez e confiabilidade do modelo, é crucial realizar o teste em vários conjuntos de dados de teste. Isso ajuda a identificar quaisquer tendências ou vieses específicos do conjunto de dados. Além disso, o teste de validação cruzada pode fornecer uma estimativa mais precisa do desempenho do modelo em dados não vistos.

**Parágrafo 3:** As fórmulas para essas métricas são as seguintes:

* Precisão: TP / (TP + FP)
* Recall: TP / (TP + FN)
* Pontuação F1: 2TP / (2TP + FP + FN)
* ROC-AUC: Área sob a curva Característica de Operação do Receptor

Item do edital: Avaliação de modelos de classificação: validação

**Parágrafo 1:**

A validação é crucial na avaliação de modelos de classificação, verificando seu desempenho em dados não vistos. Compara as previsões do modelo com os rótulos verdadeiros, usando métricas como precisão, recall e pontuação F1. Para calcular a precisão, dividimos o número de previsões corretas pelo número total de previsões.

**Parágrafo 2:**

Existem vários tipos de validação, incluindo validação cruzada, validação de conjunto de teste e validação em tempo real. A validação cruzada divide os dados em conjuntos de treinamento e teste, repetidamente treinando e testando o modelo em diferentes subconjuntos para obter uma estimativa mais confiável do desempenho. A validação do conjunto de teste envolve reservar um conjunto específico de dados para teste após o treinamento do modelo.

**Parágrafo 3:**

A matriz de confusão resume o desempenho do modelo, exibindo os verdadeiros positivos (TP), falsos negativos (FN), falsos positivos (FP) e verdadeiros negativos (TN). Usando essas métricas, podemos calcular a pontuação F1, que pondera precisão e recall: Pontuação F1 = (2 \* Precisão \* Recall) / (Precisão + Recall). Ao avaliar modelos de classificação, a validação garante que o desempenho medido represente com precisão o desempenho do modelo em dados do mundo real.

Item do edital: Avaliação de modelos de classificação: validação cruzada

**Parágrafo 1:** A validação cruzada é uma técnica de avaliação de modelo que divide os dados do conjunto de treinamento em várias dobras. Cada dobra é usada como conjunto de teste enquanto as outras dobras são usadas para treinar o modelo. O processo é repetido para cada dobra, e o desempenho do modelo é medido como a média dos resultados de todas as dobras.

**Parágrafo 2:** Existem diferentes métodos de validação cruzada, incluindo validação cruzada K-fold (onde os dados são divididos em K dobras iguais) e validação cruzada leave-one-out (onde cada instância de dados é usada uma vez como conjunto de teste). A escolha do método depende do tamanho do conjunto de dados e da especificidade do problema de classificação.

**Parágrafo 3:** A fórmula geral para calcular a acurácia da validação cruzada é:

Accuracy = (1/K) \* Σ(i=1 to K) (Number of correct predictions in fold i / Total number of predictions in fold i)

onde:

* K é o número de dobras
* Número de previsões corretas na dobra i é o número de previsões que são iguais às verdadeiras classes para as instâncias de dados na dobra i
* Número total de previsões na dobra i é o número de instâncias de dados na dobra i

Item do edital: Avaliação de modelos de classificação: métricas de avaliação - matriz de confusão

**Parágrafo 1:** A matriz de confusão é uma ferramenta valiosa para avaliar modelos de classificação, fornecendo uma visão abrangente do desempenho do modelo. Ela resume os resultados da classificação em uma tabela, mostrando quantos exemplos foram corretamente classificados como positivos verdadeiros (TP), negativos verdadeiros (TN), falsos positivos (FP) e falsos negativos (FN).

**Parágrafo 2:** As métricas de avaliação derivadas da matriz de confusão incluem:

* **Acurácia:** Proporção de exemplos corretamente classificados (TP+TN)/(TP+TN+FP+FN).
* **Precisão:** Proporção de exemplos positivos previstos que são verdadeiramente positivos (TP)/(TP+FP).
* **Revocação (Recall):** Proporção de exemplos positivos verdadeiros previstos (TP)/(TP+FN).
* **Pontuação F1:** Média harmônica de precisão e revocação, que pondera igualmente ambos os aspectos.

**Parágrafo 3:** Interpretar uma matriz de confusão requer atenção especial aos seguintes fatores:

* **Classificação Incorreta:** As células FP e FN representam erros de classificação cometidos pelo modelo.
* **Desequilíbrio de Classe:** Se as classes forem desequilibradas (uma classe tem muito mais exemplos), a acurácia pode ser enganosa e métricas como precisão e revocação devem ser priorizadas.
* **Limiar de Classificação:** O limiar usado para classificar exemplos como positivos ou negativos pode afetar as métricas de avaliação.

Item do edital: Avaliação de modelos de classificação: acurácia

A precisão mede a proporção de previsões corretas feitas por um modelo de classificação. É calculada como o número de previsões corretas dividido pelo número total de previsões. Matematicamente, é representado como:

Precisão = (Verdadeiros Positivos + Verdadeiros Negativos) / (Verdadeiros Positivos + Verdadeiros Negativos + Falsos Positivos + Falsos Negativos)

Um modelo de classificação com alta precisão tem uma propensão menor de fazer previsões incorretas. No entanto, é importante notar que a precisão pode ser enganosa se a distribuição de classes for desequilibrada. Por exemplo, se a maioria das instâncias pertencer a uma classe, um modelo que prevê essa classe para todas as instâncias terá alta precisão, embora não seja preditivo.

Item do edital: Avaliação de modelos de classificação: precisão

**Parágrafo 1:**

A precisão é uma métrica fundamental para avaliar modelos de classificação, medindo a proporção de previsões corretas entre todas as previsões feitas. É calculada como:

Precisão = TP / (TP + FP)

Onde:

* TP é o número de verdadeiros positivos (previsões corretas de uma classe específica)
* FP é o número de falsos positivos (previsões incorretas de uma classe específica)

**Parágrafo 2:**

Um modelo de classificação com alta precisão indica que faz poucas previsões incorretas da classe específica. Em contraste, um modelo com baixa precisão faz muitas previsões incorretas, levando a uma taxa mais alta de falsos positivos. Se um modelo tem uma precisão de 1, significa que faz todas as previsões corretas para a classe específica.

**Parágrafo 3:**

A precisão é particularmente importante quando o custo de falsos positivos é alto. Por exemplo, em um sistema de detecção de fraude, falsos positivos podem resultar em investigações desnecessárias ou bloqueio injusto de contas. Nesses casos, um modelo com alta precisão é crucial para minimizar o impacto negativo dos falsos positivos.

Item do edital: Avaliação de modelos de classificação: revocação

**Parágrafo 1:**

A revocação é uma medida de desempenho que avalia a capacidade de um modelo de classificação de identificar corretamente instâncias positivas (verdadeiro positivo), entre todas as instâncias positivas reais. Ela é calculada como o número de verdadeiros positivos dividido pelo número total de instâncias positivas na amostra.

**Parágrafo 2:**

A fórmula para a revocação é: R = TP / (TP + FN)

Onde:

* R = Revocação
* TP = Verdadeiros positivos
* FN = Falsos negativos

**Parágrafo 3:**

Um modelo com alta revocação é capaz de identificar com sucesso a maioria das instâncias positivas, minimizando os falsos negativos. Isso é importante em cenários como detecção de fraudes ou diagnóstico médico, onde é crucial não perder nenhuma instância positiva genuína. A combinação de alta revocação com outras métricas, como precisão e F1-score, fornece uma avaliação abrangente do desempenho do modelo de classificação.

Item do edital: Avaliação de modelos de classificação: F1-score

**Parágrafo 1:** O F1-score é uma medida de desempenho frequentemente utilizada para avaliar modelos de classificação binária. Ele combina precisão e revocação em uma única métrica, variando de 0 a 1. Um F1-score de 1 indica que o modelo classifica perfeitamente todos os exemplos, enquanto um F1-score de 0 indica que o modelo não classifica corretamente nenhum exemplo.

**Parágrafo 2:** A fórmula para calcular o F1-score é:

F1-score = 2 \* (Precisão \* Revocação) / (Precisão + Revocação)

onde Precisão é o número de verdadeiros positivos dividido pelo número de falsos positivos, e Revocação é o número de verdadeiros positivos dividido pelo número de falsos negativos.

**Parágrafo 3:** Um F1-score alto indica que o modelo é capaz de identificar corretamente tanto exemplos positivos quanto negativos. Um F1-score baixo pode sugerir que o modelo está lutando para identificar corretamente uma ou ambas as classes, ou que os dados estão desequilibrados. O F1-score é particularmente útil quando a distribuição das classes é desbalanceada, pois fornece uma medida de desempenho mais robusta do que a precisão ou revocação sozinha.

Item do edital: Avaliação de modelos de classificação: curva ROC.

**Parágrafo 1:**

A curva Característica de Operação do Receptor (ROC) é uma ferramenta visual para avaliar o desempenho dos modelos de classificação. Ela plota a taxa de verdadeiro positivo (TPR) contra a taxa de falso positivo (FPR) para diferentes limiares de classificação. A TPR representa a capacidade do modelo de identificar corretamente instâncias positivas, enquanto a FPR representa a taxa de identificar incorretamente instâncias negativas como positivas.

**Parágrafo 2:**

A curva ROC pode ser usada para comparar diferentes modelos de classificação e selecionar o melhor modelo para uma aplicação específica. Um modelo com uma curva ROC mais alta geralmente tem melhor desempenho do que um modelo com uma curva ROC mais baixa. A área sob a curva ROC (AUC) é uma métrica resumida do desempenho do modelo, variando de 0 a 1. Uma AUC mais alta indica melhor desempenho.

**Parágrafo 3:**

A equação para calcular a TPR é TPR = TP / (TP + FN), onde TP é o número de verdadeiros positivos (instâncias positivas corretamente classificadas) e FN é o número de falsos negativos (instâncias positivas incorretamente classificadas como negativas). A equação para calcular a FPR é FPR = FP / (FP + TN), onde FP é o número de falsos positivos (instâncias negativas incorretamente classificadas como positivas) e TN é o número de verdadeiros negativos (instâncias negativas corretamente classificadas).

Item do edital: Técnicas de regressão: Redes neurais para regressão

**Parágrafo 1:**

As redes neurais são modelos de aprendizado de máquina inspirados no funcionamento do cérebro humano. Elas são capazes de aprender padrões complexos a partir de dados, tornando-as uma ferramenta poderosa para resolver problemas de regressão. As redes neurais para regressão geralmente consistem em uma camada de entrada que recebe os dados de entrada, uma ou mais camadas ocultass que processam os dados e uma camada de saida que produz a previsão de regressão.

**Parágrafo 2:**

Uma técnica comum de rede neural para regressão é a rede neural feedforward. Essas redes são compostas de neurônios dispostas em camadas, com cada neurônio conectado aos neurônios da camada seguinte. Os pesos e biases dos neurônios são ajustados durante o processo de aprendizado para minimizar a função de perda entre as previsões da rede e os valores reais da amostra. A função de perda mais comumente usado em tarefas de regressão é o erro quadrado médio (MSE):

MSE = (1/n) \* ∑(y\_i - y\_pred\_i)^2

onde n é o tamanho do conjunto de dados, y\_i é o valor real e y\_pred\_i é a previsão da rede.

**Parágrafo 3:**

Outras técnicas de rede neural para regressão incluem:

* **Redes neurais convolucionais (CNNs):** Projetadas para processar dados de grade, como imagens e dados de sensores. Elas são particularmente adequadas para tarefas de regressão que envolvem extrair padrões espaciais.
* **Redes neurais recorrentes (RNNs):** Projetadas para processar dados sequenciais, como texto e dados de série temporal. Elas são particularmente adequadas para tarefas de regressão que envolvem modelar dependências temporais.
* **Redes neurais autocodificadoras:** Projetadas para aprender representações compactas e eficientes de dados. Elas podem ser usadas para tarefas de regressão sem supervisão ou como uma camada pré-treinada para redes neurais supervisionadas.

Item do edital: Árvores de decisão para regressão

**Parágrafo 1:**

As Árvores de Decisão para Regressão são um algoritmo de aprendizagem supervisionada que aproxima uma função de regressão por meio da construção de uma árvore de decisão hierárquica. Elas dividem iterativamente os dados de entrada em subconjuntos menores com base em valores de atributo, criando nós internos e nós folha. Cada nó interno representa uma decisão (ponto de divisão) e cada nó folha representa uma previsão contínua (valor de destino).

**Parágrafo 2:**

O processo de divisão é guiado por uma métrica de impureza, como o erro quadrático médio (MSE). O MSE mede a diferença entre os valores previstos e os valores reais para cada subconjunto. Os nós são divididos até que um critério de parada seja atingido, como uma profundidade máxima ou uma impureza mínima.

**Parágrafo 3:**

Para prever o valor de destino para uma nova amostra, ela é passada pela árvore de decisão, seguindo os ramos de acordo com os valores de atributo. A previsão final é o valor médio dos valores de destino nos nós folha alcançados. As fórmulas usadas no treinamento da árvore de decisão incluem:

* **MSE:** MSE = (1/n) \* Σ (y\_i - ŷ\_i)^2
* **Entropia:** H(Y) = -Σ (p\_i \* log(p\_i))
* **Gini:** G(Y) = 1 - Σ (p\_i)^2

Item do edital: Máquinas de vetores de suporte para regressão

As Máquinas de Vetores de Suporte (SVMs) para regressão são um algoritmo de aprendizado de máquina supervisionado projetado para mapear dados de entrada para valores de saída contínuos. Ao contrário das SVMs tradicionais para classificação, as SVMs para regressão usam uma função de perda diferente que permite saídas numéricas.

Matematicamente, uma SVM de regressão resolve o seguinte problema de minimização:

min(0,5||w||^2 + C \* ∑\_{i=1}^m(max(0, y\_i - f(x\_i)) - ε))^2)

Onde:

* w é o vetor de pesos
* C é o parâmetro de regularização
* m é o número de pontos de dados
* y\_i é o rótulo de saída do i-ésimo ponto de dados
* f(x\_i) é a saída prevista do modelo para o i-ésimo ponto de dados
* ε é o valor ε-insensível

Este problema de minimização visa minimizar o erro de previsão, equilibrando a complexidade do modelo (regulado pelo termo ||w||^2) com a conformidade com os dados (regulada pelo termo de perda).

Na solução ótima, a SVM de regressão encontra um hiperplano que separa os pontos de dados com base em seus valores de saída. No entanto, ao contrário da classificação, o hiperplano é construído para minimizar o erro de previsão, permitindo que alguns pontos de dados violem a margem com um custo penal.

Item do edital: Ajuste de modelos dentro e fora de amostra e overfitting.

**Ajuste de Modelos**

O ajuste de modelos envolve encontrar os valores dos parâmetros de um modelo estatístico que melhor descrevem os dados observados. Isso é feito minimizando uma função de perda, que mede a diferença entre as previsões do modelo e os valores observados. Existem dois tipos principais de ajuste de modelo: ajuste dentro da amostra e ajuste fora da amostra.

**Ajuste Dentro e Fora da Amostra**

No ajuste dentro da amostra, o modelo é ajustado usando os mesmos dados que serão usados ​​para fazer previsões. Isso pode levar ao overfitting, que ocorre quando o modelo se adapta muito aos dados específicos e perde sua capacidade de generalizar para novos dados.

O ajuste fora da amostra, por outro lado, envolve ajustar o modelo em um conjunto de dados diferente daquele que será usado para fazer previsões. Isso ajuda a evitar o overfitting e fornece uma estimativa mais precisa do desempenho do modelo.

**Overfitting**

Overfitting ocorre quando um modelo estatístico está muito próximo dos dados de treinamento e não consegue generalizar para novos dados. Isso pode ser medido pela métrica de overfitting, que é calculada como a diferença entre a precisão do modelo nos conjuntos de dados de treinamento e teste. Uma métrica de overfitting alta indica overfitting. Para evitar overfitting, é importante usar técnicas de regularização, como eliminação de recursos, seleção de modelos e validação cruzada.

Item do edital: Técnicas de agrupamento:

**Parágrafo 1**

O agrupamento é uma técnica de aprendizado de máquina não supervisionado que visa dividir um conjunto de dados em grupos distintos chamados clusters. Cada cluster contém pontos de dados semelhantes entre si e diferentes dos pontos de outros clusters. O objetivo do agrupamento é identificar padrões e estruturas ocultas nos dados sem depender de rótulos ou informações prévias.

**Parágrafo 2**

Existem vários algoritmos de agrupamento, cada um com suas vantagens e desvantagens. Alguns algoritmos comuns incluem:

* **Agrupamento K-Means:** Divide os dados em K clusters, onde K é especificado pelo usuário.
* **Agrupamento Hierárquico:** Cria uma hierarquia de clusters, permitindo que os usuários explorem diferentes níveis de granularidade.
* **Agrupamento Densidade DBSCAN:** Identifica clusters com base na densidade de pontos, permitindo encontrar clusters de formas arbitrárias.

**Parágrafo 3**

O agrupamento é amplamente utilizado em vários domínios, como:

* **Segmentação de clientes:** Identificar grupos distintos de clientes com base em seus comportamentos ou preferências.
* **Detecção de anomalias:** Encontrar pontos de dados que se desviam do comportamento normal.
* **Recomendações:** Sugerir produtos ou serviços que provavelmente interessarão aos usuários com base em seus clusters.

Item do edital: Agrupamento por partição

O agrupamento por partição é um algoritmo de agrupamento sem supervisão que visa dividir um conjunto de dados em subconjuntos ou "partições" significativas, também conhecidas como clusters. Ele funciona atribuindo inicialmente cada ponto de dados aleatoriamente a uma partição e, em seguida, iterativamente atualiza as partições movendo pontos de dados entre elas para minimizar uma função de perda que mede a dissimilaridade dentro das partições.

Matematicamente, a função de perda mais comumente usada para agrupamento por partição é a soma das distâncias quadradas entre cada ponto de dados e o centroide de sua partição, onde o centroide é o ponto médio dos pontos de dados na partição. A função de perda é dada por:

J = ∑\_{i=1}^{k} ∑\_{x ∈ C\_i} ||x - μ\_i||^2

onde:

* k é o número de partições
* C\_i é a i-ésima partição
* x é um ponto de dados em C\_i
* μ\_i é o centroide de C\_i

O algoritmo de agrupamento por partição continua iterativamente atualizando as partições até que a função de perda não possa mais ser reduzida significativamente. No final, cada ponto de dados é atribuído à partição com o centroide mais próximo.

Item do edital: Agrupamento por densidade

êle BRARY терпе терпе التس ارز⠜ 酥 Wikisource‎‎‎‎‎‎‎‎‎‎ Thermostat‎‎‎‎‎程式

Item do edital: Agrupamento hierárquico.

O agrupamento hierárquico é um método de agrupamento que constrói uma hierarquia de clusters, representando as relações entre os pontos de dados. Ele começa com cada ponto de dados em seu próprio cluster e, em seguida, iterativamente mescla os clusters mais próximos até que todos os pontos de dados estejam em um único cluster.

A medida de distância usada para determinar a semelhança entre os pontos de dados é fundamental no agrupamento hierárquico. Comumente, a distância euclidiana ou a distância de Manhattan é usada. A distância entre dois clusters é definida como a distância mínima entre quaisquer dois pontos dos clusters.

Existem dois métodos principais de agrupamento hierárquico: agrupamento de ligação única e agrupamento de ligação completa. No agrupamento de ligação única, a distância entre dois clusters é definida como a menor distância entre quaisquer dois pontos dos clusters. No agrupamento de ligação completa, a distância entre dois clusters é definida como a maior distância entre quaisquer dois pontos dos clusters.

Item do edital: Técnica de redução de dimensionalidade: Seleção de características (feature selection)

**Parágrafo 1:**

A Seleção de Características é uma técnica de redução de dimensionalidade que visa reduzir o número de características em um conjunto de dados sem comprometer significativamente sua capacidade de prever o alvo. Ela envolve selecionar um subconjunto de características que são mais relevantes para a tarefa de aprendizado de máquina. A redução da dimensionalidade pode melhorar o desempenho do modelo, reduzir o tempo de treinamento e facilitar a interpretabilidade.

**Parágrafo 2:**

Existem vários métodos de Seleção de Características, incluindo:

* **Filtro:** Avalia as características individualmente usando métricas estatísticas (por exemplo, correlação, informação mútua).
* **Wrapper:** Avalia subconjuntos de características usando um modelo de aprendizado de máquina e seleciona o subconjunto com o melhor desempenho.
* **Incorporado:** Incorpora a seleção de características no processo de aprendizado de máquina, como na regressão L1 (LASSO).

**Parágrafo 3:**

As fórmulas usadas na Seleção de Características podem variar dependendo do método usado. Por exemplo, na correlação de Pearson, a fórmula é:

corr(X\_i, y) = (cov(X\_i, y)) / (std(X\_i) \* std(y))

onde:

* X\_i é a característica i
* y é o alvo
* cov(X\_i, y) é a covariância entre X\_i e y
* std(X\_i) é o desvio padrão de X\_i
* std(y) é o desvio padrão de y

Item do edital: Técnicas de redução de dimensionalidade: análise de componentes principais (PCA – principal component analysis).

A Análise de Componentes Principais (PCA) é uma técnica de redução de dimensionalidade linear que identifica as direções de maior variância em um conjunto de dados multidimensional. Ao projetar os dados nessas direções, a PCA reduz a dimensionalidade enquanto preserva a maior parte da variância original. Isso é alcançado encontrando os autovetores e autovalores da matriz de covariância dos dados. Os autovetores representam as novas direções, enquanto os autovalores indicam a quantidade de variância explicada por cada direção.

A PCA pode ser formalizada matematicamente como:

X = UΛV'

Onde:

* X é a matriz de dados original
* U é a matriz de autovetores
* Λ é a matriz diagonal de autovalores
* V' é a transposta da matriz de autovetores

A matriz resultante X contém as projeções dos dados nas novas direções de menor dimensionalidade. A dimensão das projeções pode ser reduzida selecionando apenas os autovetores correspondentes aos autovalores maiores, que representam os componentes mais significativos.

A PCA é usada em uma ampla gama de aplicações, incluindo reconhecimento de padrões, análise de dados exploratória e compressão de dados. É particularmente útil quando os dados são altamente dimensionais e há uma forte correlação entre as variáveis originais.

Item do edital: Processamento de linguagem natural: Normalização textual

**Parágrafo 1:**

O Processamento de Linguagem Natural (PNL) é um subcampo da Inteligência Artificial que permite que os computadores compreendam e processem a linguagem humana. A normalização textual é uma etapa crucial no processamento de texto para preparar dados para várias tarefas de PNL, como classificação de texto e modelagem de linguagem. Remove variações de texto que podem dificultar o processamento, como pontuação, maiúsculas e minúsculas.

**Parágrafo 2:**

Existem diferentes técnicas para normalização textual. A tokenização divide o texto em unidades menores (tokens), enquanto a remoção de stop words remove palavras comuns que não contribuem significativamente para o significado. A redução de stemming e lematização remove sufixos ou prefixos, respectivamente, reduzindo palavras a suas formas básicas.

**Parágrafo 3:**

Fórmulas matemáticas podem ser usadas para medir a distância de Levenshtein, uma métrica de similaridade de string que quantifica o número de operações de edição necessárias para transformar uma string em outra. Outras fórmulas comuns incluem TF-IDF (Termo Frequência-Inverso Frequência Documental) e Jaccard Similarity, que medem a similaridade entre documentos ou conjuntos de palavras. Essas fórmulas são usadas para tarefas como desduplicação de texto e agrupamento de documentos.

Item do edital: Processamento de linguagem natural: stop words

**Parágrafo 1:**

As stop words são palavras comuns, como "o", "e" e "para", que aparecem frequentemente no texto, mas contribuem pouco para o seu significado. Elas podem ser uma distração durante o processamento de linguagem natural (PNL), pois podem obscurecer padrões e relacionamentos mais importantes. Por exemplo, em uma frase como "O gato preguiçoso dormiu", as stop words "o" e "para" não fornecem informações valiosas sobre o gato ou suas ações.

**Parágrafo 2:**

Para melhorar o desempenho do PNL, as stop words costumam ser removidas do texto antes do processamento. Isso pode ser feito usando uma lista pré-compilada de stop words ou por meio de algoritmos que identificam palavras que ocorrem acima de um determinado limite. Essa remoção simplifica o texto, tornando mais fácil para os algoritmos de PNL extraírem informações significativas.

**Parágrafo 3:**

Embora a remoção de stop words possa aumentar a precisão do PNL em algumas tarefas, também pode ter desvantagens. Por exemplo, em uma tarefa de reconhecimento de entidade nomeada, onde o objetivo é identificar nomes de pessoas, lugares e organizações, a remoção de stop words pode levar à perda de informações importantes. Portanto, a decisão de remover ou não as stop words deve ser tomada caso a caso, considerando os objetivos específicos da tarefa de PNL.

Item do edital: Processamento de linguagem natural: estemização

**Parágrafo 1:**

Estemização é um processo de redução de palavras à sua forma raiz ou radical. Ele remove sufixos e prefixos, revelando o núcleo semântico da palavra. Por exemplo, "correndo", "corrida" e "corredor" seriam todos reduzidos ao radical "corr". Isso ajuda na indexação e recuperação de informações, pois palavras com significados semelhantes são mapeadas para uma única representação.

**Parágrafo 2:**

Existem vários algoritmos de estemização usados, incluindo o algoritmo de Porter e o algoritmo de Lancaster. Esses algoritmos usam conjuntos de regras para remover sufixos e prefixos comuns. Por exemplo, o algoritmo de Porter remove o sufixo "-ing" de "correndo".

**Parágrafo 3:**

A estemização é uma etapa importante no pré-processamento de texto para tarefas de processamento de linguagem natural (PNL). Ao reduzir as palavras às suas formas raiz, os algoritmos de estemização ajudam a identificar relações semânticas entre palavras e melhorar a precisão de tarefas como classificação de texto, extração de informações e recuperação de informações.

Item do edital: Processamento de linguagem natural: lematização

**Parágrafo 1:**

O Processamento de Linguagem Natural (PNL) é um campo de Inteligência artificial que permite aos dispositivos digitais compreender e processar a linguagem humana. Ele abrange uma ampla gama de tarefas, como reconhecimento de voz, síntese de fala, tradução automática e processamento de texto. O objetivo final do PNL é quebrar as barreiras da comunicação entre máquinas e seres.

**Parágrafo 2:**

Uma subárea fundamental do PNL é o aprendizado supervisionado, no qual os dispositivos aprendem a mapear padrões de entrada a saídas desejadas. Para treinamento, são usadas fórmulas como a **Função de Perda Quadrada Média (MSE)**:

MSE = (1/n) Σ (y\_i - f(x\_i))^2

onde:

* *n* representa o número de pares de entrada-saídas
* *y\_i* representa a *i\_iésima* amostra de entrada
* *f(x\_i)* representa a previsão para a *i\_iésima* amostra de entrada

**Parágrafo 3:**

O PNL tem inúmeras implicações práticas. Ele apriмора a pesquisa na Internet, tornando os mecanismos de busca mais precisos. Ele ajuda no atendimento ao cliente, permitindo que os chatbots compreendam e respondam a consultas. Além do mais, o PNL capacita aplicativos de tradução de linguagem, abrindo possibilidades para comunicação global. Com o avanço contínuo da tecnologia, o PNL promete moldar ainda mais a forma como interagimos com nossos dispositivos.

Item do edital: Processamento de linguagem natural: análise de frequência de termos

**Resumo de Processamento de Linguagem Natural: Análise de Frequência de Termos**

A análise de frequência de termos é uma técnica de processamento de linguagem natural (PNL) que analisa a frequência de ocorrência de palavras ou termos específicos em um texto. Essa técnica é usada para identificar palavras-chave, entender o contexto e extrair informações importantes.

Para calcular a frequência de um termo, usamos a seguinte fórmula:

Frequência = (Número de ocorrências do termo no texto) / (Número total de palavras no texto)

Aqui estão os passos envolvidos na análise de frequência de termos:

* **Pré-processamento:** Remover pontuação, números e palavras de parada comuns.
* **Tokenização:** Dividir o texto em tokens individuais (palavras ou termos).
* **Stemming ou Lematização:** Reduzir as palavras às suas formas básicas (por exemplo, "correr", "correndo" -> "correr").
* **Contagem de frequência:** Contar as ocorrências de cada termo distinto.
* **Análise:** Interpretar os resultados da contagem de frequência para extrair insights e identificar padrões significativos.

A análise de frequência de termos é usada em vários aplicativos de PNL, como:

* Mineração de texto
* Extração de informações
* Classificação de texto
* Geração de linguagem

Item do edital: Rotulação de partes do discurso: part-of-speech tagging

**Parágrafo 1:**

A rotulação de partes do discurso (POS) é a tarefa de atribuir etiquetas gramaticais a cada palavra de um texto. Essas etiquetas indicam o papel gramatical da palavra (por exemplo, substantivo, verbo, adjetivo) e fornecem informações valiosas para o processamento de linguagem natural (PNL).

**Parágrafo 2:**

Os modelos de rotulação de POS geralmente usam técnicas de aprendizado de máquina que são treinadas em grandes conjuntos de dados rotulados manualmente. Esses modelos aprendem a identificar padrões na sequência de palavras e atribuir as tags POS apropriadas. A fórmula geral para a pontuação de um modelo de POS é:

F1 = (2 \* Precisão \* Revocação) / (Precisão + Revocação)

onde precisão é a proporção de tags POS corretas e revocação é a proporção de tags POS corretas entre as tags esperadas.

**Parágrafo 3:**

A rotulação de POS é crucial para muitas tarefas de PNL, incluindo análise sintática, análise semântica, recuperação de informação e tradução automática. Fornece uma base de conhecimento sobre a estrutura gramatical de uma frase, facilitando o entendimento e a manipulação do texto por computadores.

Item do edital: Modelos de representação de texto: N-gramas

**Parágrafo 1:**

Os modelos N-grama representam sequências de tokens de texto (palavras, caracteres, etc.) como vetores numéricos. Um N-grama de ordem n é uma sequência de n tokens consecutivos. Por exemplo, a frase "Eu gosto de maçãs" pode ser representada como o seguinte trigrama (N-grama de ordem 3): Eu-gosto-de. Os vetores de N-grama são frequentemente criados contando a frequência de ocorrência de cada N-grama em um corpus de texto.

**Parágrafo 2:**

A fórmula para calcular a frequência de um N-grama X numa sequência de texto T é:

Freq(X, T) = (# de ocorrências de X em T) / (# total de N-gramas em T)

**Parágrafo 3:**

Os modelos de N-grama são simples e fáceis de implementar, mas podem capturar informações locais sobre a estrutura do texto. Eles são comumente usados no processamento de linguagem natural (PLN) para tarefas como modelagem de linguagem, análise sintática e classificação de texto. No entanto, eles podem ser propensos a sparsidade (com muitos N-gramas com frequência zero) e não conseguem capturar dependências de longo alcance entre tokens.

Item do edital: modelos vetoriais de palavras: CBOW

ライフ=========ή

sustenance sustenance e n t s t t == == == == == === === === ======== =========== ====== ======= ===

Item do edital: modelos vetoriais de palavra: Skip-Gram

**Skip-Gram**

O modelo Skip-Gram, proposto no artigo do Word2Vec, é um modelo de incorporação de palavras que visa capturar as relações contextuais entre as palavras. Ele constrói vetores de representação para palavras individuais, onde cada vetor representa a distribuição de contexto de uma palavra.

Formalmente, dado um texto corpus, cada palavra **w** é tratada como um vetor binário de tamanho **|V|** (onde **|V|** é o tamanho do vocabulário). O modelo Skip-Gram define uma função de pontuação **f(w, c)** que mede a probabilidade de uma palavra de contexto **c** aparecer em um determinado contexto de uma palavra **w**. O modelo é treinado para maximizar a soma das pontuações de log-verossimilhança sobre todos os pares de palavras e seus respectivos contextos:

L = ∑(log f(w, c))

**Implementação**

O modelo Skip-Gram é implementado usando redes neurais de duas camadas. A camada de entrada codifica a palavra alvo **w** como um vetor denso de tamanho **d**. A camada de saída é uma camada softmax que prevê a distribuição de contexto de **w** sobre todas as palavras no vocabulário. Os pesos da rede são otimizados usando descida de gradiente estocástica para minimizar a perda de log-verossimilhança.

Item do edital: modelos vetoriais de palavra: GloVe

Os modelos vetoriais de palavras, como o GloVe (Global Vectors for Word Representation), visam capturar o significado e as relações entre as palavras convertendo-as em vetores numéricos. Assim, cada palavra é representada por um vetor de valores reais que capta seu significado semântico.

O GloVe foi desenvolvido por Pennington et al. (2014) e combina as vantagens dos modelos de co-ocorrência e de fatoração matricial para criar vetores de palavras. Ele começa construindo uma matriz de co-ocorrência que conta o número de vezes que cada par de palavras aparece em uma determinada janela de contexto. Essa matriz é então fatorada em uma matriz de vetores de palavras e uma matriz de vetores de contexto usando a fatoração singular de valores.

O objetivo da fatoração é minimizar a seguinte função de perda:

J(V, C) = ∑(i, j) w(i, j) \* (V\_i^T \* C\_j - log(X\_i, j))^2

onde:

* V é a matriz de vetores de palavras
* C é a matriz de vetores de contexto
* X é a matriz de co-ocorrência
* w(i, j) é um peso que penaliza pares de palavras frequentes

Item do edital: modelos vetoriais de documentos: booleano

Os modelos vetoriais de documentos (VSMs) são representações matemáticas de documentos como vetores, com cada elemento representando a frequência ou importância de um termo específico no documento. Esses modelos são amplamente utilizados em recuperação de informações e processamento de linguagem natural.

Um VSM básico representa um documento como um vetor de comprimento fixo, com cada elemento correspondendo a um termo no vocabulário. Cada elemento é preenchido com a frequência do termo (TF) ou frequência do termo ponderada (TF-IDF), que considera tanto a frequência quanto a importância do termo no corpus de documentos.

Modelos vetoriais mais avançados podem incorporar outras informações, como a ordem das palavras ou a proximidade entre os termos. Esses modelos geralmente envolvem o uso de algoritmos de processamento de linguagem natural para extrair recursos semânticos do texto, que são então usados para construir os vetores de documentos.

Item do edital: modelos vetoriais de documentos: TF

**Modelo de Termos em Frequência (TF)**

O modelo de TF mede a frequência de um termo no documento. Ele quantifica quantas vezes um termo específico aparece em um documento, normalizado pelo comprimento do documento.

**Fórmula:**

TF(t, d) = f(t, d) / N(d)

Onde:

* f(t, d) é a frequência do termo t no documento d
* N(d) é o número total de tokens no documento d

**Interpretação:**

Valores altos de TF indicam que o termo é proeminente naquele documento em particular, enquanto valores menores sugerem que o termo é menos importante. O modelo de TF é frequentemente usado como uma medida de relevância de um termo para um documento e pode ser usado na busca de informações e outros aplicativos de processamento de texto.

Item do edital: modelos vetoriais de documentos: TF-IDF

**Parágrafo 1:**

Os modelos vetoriais de documento (DVM) representam documentos como vetores em um espaço multidimensional, onde cada dimensão corresponde a um termo específico. Os DVMs usam o peso termo-frequência (TF) para quantificar a ocorrência de um termo em um documento e o peso inverso da frequência de documento (IDF) para normalizar a importância do termo em todo o corpus.

**Parágrafo 2:**

A fórmula para TF-IDF é:

TF-IDF(t, d) = TF(t, d) \* IDF(t)

Onde:

* TF(t, d) é a frequência do termo t no documento d
* IDF(t) é o inverso da frequência do documento de t, calculado como:

IDF(t) = log(N / df(t))

* N é o número total de documentos no corpus
* df(t) é o número de documentos que contêm o termo t

**Parágrafo 3:**

Os modelos vetoriais de documento são amplamente usados em áreas como recuperação de informações, processamento de linguagem natural e aprendizado de máquina. O TF-IDF pondera os termos de forma a destacar aqueles que são frequentes em um documento individual, mas raros no corpus como um todo, melhorando a eficácia de tarefas como classificação e agrupamento de documentos.

Item do edital: modelos vetoriais de documentos: média de vetores de palavras

**Parágrafo 1:**

A média de vetores é um cálculo que resulta em um único vetor que representa o "vetor médio" de um conjunto de vetores. Ele fornece um valor central que resume as características comuns dos vetores dados. A fórmula para calcular a média de vetores é:

v\_média = (1/n) \* (v1 + v2 + ... + vn)

Onde:

* v\_média é o vetor médio
* n é o número de vetores
* v1, v2, ..., vn são os vetores individuais

**Parágrafo 2:**

A média de vetores tem várias propriedades. Em primeiro lugar, ela sempre residirá dentro do espaço abrangido pelos vetores originais. Em segundo lugar, ela minimiza a soma das distâncias quadradas entre cada vetor original e o vetor médio. Isso significa que a média de vetores é o vetor que está mais próximo (em termos de distância quadrada) de todos os outros vetores do conjunto.

**Parágrafo 3:**

A média de vetores é uma ferramenta valiosa em várias aplicações. Por exemplo, ela pode ser usada para encontrar o centroide de um conjunto de pontos ou para determinar a direção média de um conjunto de forças. Além disso, a média de vetores é usada em algoritmos de aprendizado de máquina, como o algoritmo k-medias, para agrupar dados em clusters.

Item do edital: modelos vetoriais de documentos: Paragraph Vector

**Parágrafo 1:** Os modelos vetoriais de documentos, como o Paragraph Vector (PV), representam documentos como vetores numéricos que capturam seu significado semântico. O PV é um modelo de aprendizado de máquina que cria esses vetores para passagens de texto. Ele mapeia as palavras em cada parágrafo para um espaço vetorial de dimensão fixa, onde palavras similares estão próximas umas das outras.

**Parágrafo 2:** Formalmente, o PV é definido como uma função que mapeia um parágrafo p para um vetor v no espaço vetorial:

v = f(p)

A função f é uma rede neural que é treinada em um grande corpus de texto. Ela aprende a representar o significado de cada parágrafo por meio de suas palavras e contexto.

**Parágrafo 3:** Os vetores de parágrafo resultantes podem ser usados para uma variedade de tarefas de processamento de linguagem natural, como recuperação de informações, resumo de texto e classificação de documentos. Eles fornecem uma representação densa e semântica do conteúdo do parágrafo, facilitando seu processamento e análise. Além disso, os vetores de parágrafo podem ser facilmente comparados usando métricas de similaridade, o que os torna adequados para tarefas como busca de similaridade de parágrafo.

Item do edital: Métricas de similaridade textual - similaridade do cosseno

**Parágrafo 1:**

A similaridade do cosseno é uma métrica de similaridade textual que calcula o ângulo entre dois vetores. Para vetores (x) e (y) no espaço vetorial, a similaridade do cosseno é definida como:

$$cos(x, y) = \frac{x \cdot y}{||x|| \cdot ||y||}$$

onde (x \cdot y) é o produto escalar e (||x||) e (||y||) são as normas euclidiana de (x) e (y), respectivamente.

**Parágrafo 2:**

A similaridade do cosseno varia entre -1 e 1. Um cosseno de 1 indica vetores idênticos, enquanto um cosseno de -1 indica vetores opostos. Cossenos próximos de 0 indicam vetores quase ortogonais, o que implica pouca semelhança.

**Parágrafo 3:**

A similaridade do cosseno é amplamente utilizada em recuperação de informações e processamento de linguagem natural. É particularmente eficaz em documentos esparsos, onde a maioria dos elementos vetoriais é zero. Isso ocorre porque a similaridade do cosseno ignora elementos zero, concentrando-se apenas nos elementos comuns entre os vetores.

Item do edital: Métricas de similaridade textual distância euclidiana

**Parágrafo 1:**

As métricas de similaridade textual destinam-se a medir a semelhança entre dois textos. A distância euclidiana é uma métrica de similaridade comumente usada que calcula a distância entre dois vetores de tamanho n, onde n representa o número de características usadas para representar os textos. Cada característica é um valor numérico que descreve um aspecto específico do texto, como a frequência da ocorrência de uma determinada palavra.

**Parágrafo 2:**

A fórmula da distância euclidiana para dois vetores x = (x1, x2, ..., xn) e y = (y1, y2, ..., yn) é dada por:

d(x, y) = sqrt((x1 - y1)² + (x2 - y2)² + ... + (xn - yn)²)

Quanto menor for o valor da distância euclidiana, mais semelhantes serão os dois textos. Um valor de zero indica que os textos são iguais, enquanto um valor maior indica que são mais diferentes.

**Parágrafo 3:**

A distância euclidiana é uma métrica simples e eficiente de calcular. No entanto, ela pode não ser adequada para todos os tipos de dados de texto. Por exemplo, ela não considera a ordem das palavras ou a estrutura gramatical dos textos. Para aplicações onde essas características são importantes, outras métricas de similaridade, como a Similaridade Jaccard ou o Coeficiente de Dice, podem ser mais adequadas.

Item do edital: Métricas de similaridade textual similaridade de Jaccard

**Métricas de Similaridade Textual**

**Similaridade de Jaccard**

A similaridade de Jaccard mede a semelhança entre dois conjuntos calculando a razão do tamanho da interseção deles pelo tamanho da união deles. No contexto da similaridade textual, a interseção é o número de palavras que aparecem em ambos os textos e a união é o número de palavras exclusivas em ambos os textos.

**Fórmula:**

Similaridade de Jaccard = Interseção / União

Item do edital: Métricas de similaridade textual distância de Manhattan

**Parágrafo 1:** A distância de Manhattan é uma métrica de similaridade textual que mede a diferença entre dois textos calculando a soma das diferenças absolutas nos valores dos recursos entre os textos. Funciona atribuindo um recurso a cada posição no texto e calculando a diferença entre os valores dos recursos na mesma posição nos dois textos. Quanto maior a distância de Manhattan, menor a similaridade entre os textos.

**Parágrafo 2:** A fórmula da distância de Manhattan é a seguinte:

d(X, Y) = Σ|x\_i - y\_i|

Onde:

* X e Y são os dois textos
* x\_i e y\_i são os valores dos recursos para a posição i em X e Y, respectivamente

**Parágrafo 3:** A distância de Manhattan é uma métrica simples e fácil de calcular, adequada para textos curtos. No entanto, pode não ser tão eficaz para textos mais longos, pois não considera a ordem das palavras ou a estrutura do texto. Além disso, o valor das diferenças absolutas pode ser exagerado quando os textos são muito diferentes, o que afeta a precisão da métrica.

Item do edital: Métricas de similaridade textual coeficiente de Dice.

**Coeficiente de Dice**

O coeficiente de Dice é uma métrica de similaridade textual que mede o grau de sobreposição entre dois conjuntos de elementos. Ele calcula a proporção do número de elementos comuns aos dois conjuntos para o número total de elementos em ambos os conjuntos. É comumente usado para comparar textos, conjuntos de palavras ou outras coleções de elementos.

**Fórmula**

O coeficiente de Dice é definido pela seguinte fórmula:

Dice(A, B) = 2 \* |A ∩ B| / (|A| + |B|)

onde:

* A e B são os dois conjuntos a serem comparados
* |A ∩ B| é o número de elementos comuns a A e B
* |A| é o número de elementos no conjunto A
* |B| é o número de elementos no conjunto B

**Interpretação**

O coeficiente de Dice varia de 0 a 1, onde:

* 0 indica que não há sobreposição entre os dois conjuntos
* 1 indica que os dois conjuntos são idênticos

Valores mais altos indicam maior similaridade, enquanto valores mais baixos indicam menor similaridade. O coeficiente de Dice é frequentemente usado em aplicações como busca de informações, processamento de linguagem natural e reconhecimento de padrões.

Item do edital: Redes neurais convolucionais

**Parágrafo 1:** As Redes Neurais Convolucionais (CNNs) são um tipo de rede neural profunda especializada em processamento de dados estruturados, como imagens e sinais. Elas são compostas por camadas de filtros convolucionais, que aplicam operações matemáticas (como a convolução) a dados de entrada para extrair padrões e características. A fórmula de convolução é:

C = F ⊗ I

Onde:

* C é a saída da convolução
* F é o filtro convolucional
* ⊗ é a operação de convolução
* I é a entrada

**Parágrafo 2:** As CNNs usam operações de agrupamento (pooling) após as camadas convolucionais para reduzir a dimensionalidade e melhorar a robustez. Existem diferentes tipos de agrupamento, como agrupamento máximo e médio. As CNNs também podem incluir camadas totalmente conectadas, que classificam ou regridem os dados extraídos pelas camadas convolucionais.

**Parágrafo 3:** As CNNs são amplamente utilizadas em visão computacional, processamento de linguagem natural e processamento de sinais. Elas são particularmente eficazes em tarefas como reconhecimento de objetos, detecção semântica e análise de séries temporais. As CNNs revolucionaram esses campos e continuam a ser uma área ativa de pesquisa e desenvolvimento.

Item do edital: Redes neurais recorrentes.

**Parágrafo 1:**

Redes neurais recorrentes (RNNs) são um tipo de rede neural projetada para processar dados sequenciais. Elas são capazes de lembrar informações de entradas anteriores, o que as torna adequadas para tarefas como processamento de linguagem natural, análise de séries temporais e reconhecimento de padrões. As RNNs operam com base no princípio de que o estado atual da rede é dependente de seus estados anteriores.

**Parágrafo 2:**

Uma RNN básica consiste em uma célula de memória, que atualiza seu estado interno a cada passo de tempo, e uma função de saída, que mapeia o estado atual para uma saída. A célula de memória pode ser implementada usando estruturas como unidades de memória de longo prazo (LSTMs) ou redes de memória a curto prazo (STMs), que permitem que a rede aprenda dependências de longo prazo em dados sequenciais.

**Parágrafo 3:**

As RNNs podem ser treinadas usando algoritmos de propagação para trás através do tempo (BPTT), que calculam os gradientes dos pesos da rede com base nas entradas e saídas sequenciais fornecidas. O treinamento de RNNs pode ser desafiador devido ao problema do gradiente de desaparecimento, onde os gradientes se tornam exponencialmente pequenos ou grandes à medida que o comprimento da sequência aumenta. Técnicas como corte de gradiente e aceleração de momento podem ser usadas para mitigar esse problema.

Item do edital: Scikit-learn

Scikit-learn é uma biblioteca Python de código aberto projetada para aprendizado de máquina. Ela fornece uma ampla gama de algoritmos e ferramentas de aprendizado de máquina, incluindo classificação, regressão, agrupamento e redução de dimensionalidade. Alguns dos algoritmos de classificação mais populares em Scikit-learn incluem:

* **Regressão logística:** Sigmoid(w\*x)
* **Árvores de decisão:** C(T, D) = I(D) - \sum\_{t \in T} I(D\_t)
* **Máquinas de vetores de suporte:** d\_+(x\_i, x\_j) + d\_-(x\_i, x\_j) = \| x\_i - x\_j \|\_2^2
* **Florestas aleatórias:** P(c|x) = \frac{1}{n} \sum\_{i=1}^n P(c|x\_i)$

Scikit-learn também oferece várias métricas de avaliação para medir o desempenho de modelos de aprendizado de máquina, como precisão, recall e pontuação F1. A biblioteca possui uma interface de usuário (UI) fácil de usar que permite que os usuários treinem, avaliem e implantem modelos rapidamente.

Além de algoritmos e métricas, Scikit-learn fornece recursos adicionais para processamento de dados, seleção de recursos e ajuste de modelo. Isso o torna uma ferramenta abrangente para cientistas de dados e profissionais de aprendizado de máquina.

Item do edital: TensorFlow

TensorFlow é uma biblioteca de software de código aberto para aprendizado de máquina e redes neurais. Foi desenvolvido pelo Google e é popular por sua versatilidade e eficiência. O TensorFlow permite que os desenvolvedores criem e treinem modelos de aprendizado de máquina usando uma abordagem baseada em gráfico, onde os dados fluem por meio de uma rede de nós de operações.

Os nós representam operações matemáticas, como adição, multiplicação ou ativações não lineares, enquanto as arestas representam os dados que fluem entre eles. Por exemplo, em uma rede neural simples para classificação de imagens, o nó de entrada pode receber uma imagem, e os nós subsequentes podem aplicar várias convoluções, operações de agrupamento e camadas totalmente conectadas para extrair recursos e fazer uma previsão.

O fluxo de dados no TensorFlow é representado por tensores, arrays multidimensionais que contêm dados numéricos. Os tensores podem ter várias dimensões, como (batch\_size, altura, largura, canais) para imagens ou (batch\_size, sequência\_comprimento) para dados de texto. As operações envolvendo tensores são expressas como equações matemáticas ou funções de biblioteca, e o TensorFlow gerencia automaticamente o cálculo e o fluxo de dados durante o treinamento e a inferência.

Item do edital: PyTorch

PyTorch é uma biblioteca de aprendizado de máquina de código aberto baseada em Python desenvolvida pelo Facebook AI Research. Ela fornece um conjunto abrangente de ferramentas e funções para o desenvolvimento e treinamento de modelos de aprendizado de máquina. A estrutura é projetada para ser flexível e escalável, permitindo aos usuários criar e treinar modelos complexos em diferentes dispositivos, incluindo CPUs e GPUs.

Um recurso fundamental do PyTorch é seu uso do cálculo automático, que permite a computação de gradientes em tempo de execução por meio da propagação reversa. Isso torna o treinamento de modelos conveniente e eficiente, pois elimina a necessidade de derivadas manuais. A fórmula geral para propagação reversa é:

dL/dw = (dL/dy) \* (dy/dw)

onde:

* dL/dw é o gradiente da perda em relação ao peso w
* dL/dy é o gradiente da perda em relação à saída y
* dy/dw é o gradiente da saída em relação ao peso w

Isso permite que os modelos aprendam com seus erros e atualizem seus pesos conforme necessário.

Além do cálculo automático, o PyTorch oferece uma ampla gama de funções e módulos para tarefas comuns de aprendizado de máquina, como processamento de imagens, processamento de linguagem natural e visão computacional. Isso simplifica o desenvolvimento de modelos e acelera o processo de treinamento, permitindo que os usuários se concentrem nos aspectos específicos de seus problemas de aprendizado de máquina.

Item do edital: Keras

Keras é uma biblioteca de aprendizado profundo de alto nível executada em TensorFlow ou Theano. Ele foi projetado para ser fácil de usar, modular e extensível, tornando-o adequado para uma ampla gama de aplicativos de aprendizado profundo. Ao contrário do TensorFlow, que requer um conhecimento profundo de seus gráficos de fluxo de dados e operações, o Keras abstrai esses conceitos por trás de uma API mais simples.

A API Keras é construída em torno do conceito de modelos, que representam um fluxo de dados de entrada para dados de saída. Os modelos são construídos conectando camadas, que são blocos de construção que realizam operações específicas nos dados. O Keras fornece uma ampla gama de camadas pré-construídas para tarefas comuns, como camadas convolucionais, camadas recorrentes e camadas totalmente conectadas. Além disso, o Keras permite que os usuários definam suas próprias camadas personalizadas.

O Keras é popular entre os cientistas de dados e profissionais de aprendizado de máquina devido à sua facilidade de uso e recursos poderosos. Ele oferece várias vantagens, incluindo:

* **Fácil de usar:** A API intuitiva do Keras torna-o acessível mesmo para iniciantes em aprendizado profundo.
* **Modular:** Os modelos Keras são construídos conectando camadas, o que torna fácil personalizar e experimentar diferentes arquiteturas.
* **Extensível:** O Keras permite que os usuários definam suas próprias camadas personalizadas, estendendo suas funcionalidades para problemas específicos.