Item do edital: Ingestão de dados estruturados

**Ingestão de Dados Estruturados**

A ingestão de dados estruturados envolve a importação de dados organizados em um formato padronizado para um sistema de gerenciamento de dados. Esses dados são caracterizados por:

* **Estrutura:** Organização em colunas e linhas predefinidas
* **Metadados:** Descrições da estrutura e dos tipos de dados
* **Valores distintos:** Sem valores duplicados dentro de uma coluna

**Processos de Ingestão**

A ingestão geralmente envolve as seguintes etapas:

* **Extração:** Recuperação de dados de fontes originais
* **Limpeza:** Remoção de dados duplicados, tratamento de valores ausentes e correção de erros
* **Transformação:** Alteração da estrutura ou formato dos dados para atender aos requisitos do sistema de destino
* **Carregamento:** Importação dos dados limpos e transformados para o sistema de destino

**Ferramentas de Ingestão**

Várias ferramentas podem ser usadas para ingestão de dados estruturados, incluindo:

* **ETL (Extração, Transformação, Carregamento):** Ferramentas que automatizam todo o processo de ingestão
* **APIs de Ingestão:** Fornecidas por sistemas de destino para importação direta de dados
* **Integrações Pré-Construídas:** Conexões pré-configuradas entre fontes de dados e sistemas de destino

**Fórmulas**

Existem várias fórmulas usadas para cálculos relacionados à ingestão de dados estruturados:

* **Taxa de Extração:** Número de registros extraídos da fonte por unidade de tempo
* **Taxa de Limpeza:** Número de registros limpos por unidade de tempo
* **Taxa de Transformação:** Número de registros transformados por unidade de tempo
* **Taxa de Carregamento:** Número de registros carregados no sistema de destino por unidade de tempo

**Benefícios**

A ingestão de dados estruturados oferece vários benefícios:

* **Análise Consistente:** Dados padronizados permitem análises consistentes e precisas
* **Tomada de Decisão Aprimorada:** Dados estruturados facilitam a descoberta de insights por meio de análises e inteligência de negócios
* **Integridade de Dados:** A estrutura e os metadados ajudam a garantir a integridade e a confiança dos dados
* **Eficiência:** Os processos automatizados de ingestão economizam tempo e esforço

Item do edital: Ingestão de dados semiestruturados

**Ingestão de Dados Semiestruturados**

Dados semiestruturados são aqueles com uma estrutura flexível, permitindo valores ausentes, ordem variável e diferentes tipos de dados dentro de um mesmo campo. Ingerir esses dados requer abordagens especializadas:

**Métodos de Ingestão:**

* **Mapeamento de Esquemas:** Define um esquema flexível que mapeia os dados recebidos para os campos correspondentes.
* **Análise de Texto:** Analisa os dados textuais usando processamento de linguagem natural (PNL), extraindo informações estruturadas.
* **Modelos de Aprendizado de Máquina:** Utilizam algoritmos de aprendizado de máquina para inferir a estrutura dos dados com base em padrões observados.

**Formatos de Dados Comuns:**

* JSON (JavaScript Object Notation)
* XML (Extensible Markup Language)
* CSV (Valores Separados por Vírgulas)
* LOG (Arquivos de Registro)

**Desafios:**

* Variação estrutural: os dados semiestruturados podem ter estruturas heterogêneas, dificultando a ingestão consistente.
* Valores ausentes e nulos: dados ausentes ou nulos podem distorcer a interpretação e análise.
* Tipos de dados inconsistentes: diferentes tipos de dados podem ser armazenados no mesmo campo, exigindo conversões de tipo.
* Alta dimensionalidade: dados semiestruturados frequentemente têm um número elevado de campos, resultando em alta dimensionalidade.

**Abordagens para Superar Desafios:**

* **Pré-processamento de Dados:** Preparar os dados limpando, convertendo tipos e removendo dados redundantes.
* **Gerenciamento de Valores Ausentes:** Imputar valores ausentes usando médias, medianas ou outros métodos apropriados.
* **Normalização de Dados:** Padronizar valores para facilitar comparações e análises.
* **Redução de Dimensionalidade:** Aplicar técnicas como análise de componentes principais (PCA) ou análise de agrupamento para reduzir a dimensionalidade e melhorar o desempenho analítico.

Item do edital: Ingestão de dados não estruturados

**Ingestão de Dados Não Estruturados**

Os dados não estruturados referem-se a dados que não seguem um formato ou esquema predefinido, como texto, imagens, vídeos e dados de mídia social. A ingestão de dados não estruturados é o processo de coletar, processar e armazenar esses dados para análise e processamento.

**Etapas da Ingestão de Dados Não Estruturados:**

* **Coleta de dados:** Os dados são coletados de várias fontes, como sites, redes sociais e dispositivos IoT.
* **Pré-processamento:** Os dados são limpos, tokenizados e normalizados para remoção de ruído e preparação para processamento.
* **Armazenamento:** Os dados são armazenados em repositórios especializados, como data lakes ou bancos de dados NoSQL.
* **Indexação:** Os dados são indexados para facilitar a pesquisa e recuperação eficientes.
* **Processamento:** Os dados são analisados e processados usando técnicas de aprendizado de máquina, processamento de linguagem natural e mineração de texto.

**Ferramentas e Tecnologias:**

* Apache NiFi
* Kafka Streams
* Apache Hadoop HDFS
* Apache Spark
* Elasticsearch

**Fórmula para Quantidade de Dados Não Estruturados:**

A quantidade de dados não estruturados no mundo está crescendo a uma taxa exponencial, conforme a seguinte fórmula:

D = (2^n) \* D0

Onde:

* D é a quantidade de dados não estruturados em um determinado ano
* n é o número de anos que se passou desde o ano base
* D0 é a quantidade de dados não estruturados no ano base

Esta fórmula demonstra o crescimento exponencial dos dados não estruturados, que deve quadruplicar a cada três anos.

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

**Ingestão de Dados em Lote**

A ingestão de dados em lote envolve a coleta e o processamento de grandes volumes de dados em um único processo, em vez de processá-los individualmente. Isso pode proporcionar eficiências significativas em termos de tempo e recursos.

**Como funciona:**

* Os dados são coletados de várias fontes e armazenados temporariamente em um repositório de preparação.
* O repositório de preparação remove dados duplicados, valida e transforma os dados para garantir sua consistência e integridade.
* Após a preparação, os dados são carregados em um data warehouse ou outro destino de armazenamento de dados.

**Fórmula para calcular a eficiência da ingestão em lote:**

Eficiência = (Tempo de processamento individual x Número de registros) / Tempo de processamento em lote

**Vantagens:**

* Redução do tempo de processamento
* Otimização do uso de recursos
* Redução de erros de processamento

**Desvantagens:**

* Pode atrasar a disponibilidade dos dados em tempo real
* Requer recursos computacionais mais potentes

**Aplicações comuns:**

* Carregamento de dados de transações em data warehouses
* Processamento de logs de atividades
* Treino de modelos de aprendizado de máquina (ML)

**Considerações:**

* Dimensionar os recursos para lidar com grandes volumes de dados
* Estabelecer processos de validação de dados para garantir a precisão
* Otimizar o processo de preparação de dados para melhorar a eficiência

Item do edital: Ingestão de dados em streaming

**Ingestão de Dados em Streaming**

A ingestão de dados em streaming envolve a aquisição contínua de dados de fontes externas, processamento e armazenamento em um data lake ou warehouse em tempo real. Isso permite que as empresas analisem dados em movimento para obter insights oportunos e responder rapidamente às mudanças de negócios.

**Tipos de Fontes de Dados**

* Dispositivos IoT
* Sistemas transacionais
* Mídias sociais
* Arquivos de log
* Rastreamento de aplicativos

**Arquitetura de Ingestão**

* **Conectores:** Interface com fontes de dados para captar dados.
* **Buffer:** Armazena dados temporariamente antes do processamento.
* **Processamento:** Transforma, limpa e enriquece dados.
* **Armazenamento:** Armazena os dados processados em um data lake ou warehouse.

**Fórmulas de Latência**

* **Latência de Ingestão:** Tempo decorrido entre a geração de dados e sua chegada ao sistema de ingestão.
* **Latência de Processamento:** Tempo necessário para transformar e processar dados.
* **Latência de Armazenamento:** Tempo entre o processamento e o armazenamento dos dados.

**Vantagens da Ingestão de Dados em Streaming**

* Insights em tempo real
* Detecção rápida de anomalias
* Personalização de experiência do cliente
* Otimização de operações
* Redução de custos de armazenamento

**Desafios**

* Alta taxa de volume e velocidade de dados
* Latência imprevisível
* Complexidade de processamento em tempo real
* Custos de infraestrutura

**Melhores Práticas**

* Use vários conectores para fontes de dados heterogêneas.
* Otimize os buffers para minimizar a latência.
* Implemente processamento paralelo.
* Escolha um data lake ou warehouse projetado para dados em streaming.
* Monitore continuamente o desempenho da ingestão.

Item do edital: Armazenamento de big data

**Armazenamento de Big Data**

O armazenamento de big data é uma tecnologia que gerencia conjuntos de dados maciços e complexos que excedem a capacidade de processamento e armazenamento de sistemas tradicionais.

**Características dos Dados em Grande Escala**

* Volume: Quantidade imensa de dados, medidos em terabytes ou petabytes.
* Variedade: Diferentes formatos de dados, como texto, imagens, vídeos e dados estruturados.
* Velocidade: Geração e processamento rápidos de dados.
* Veracidade: Precisão e consistência dos dados.
* Valor: Insights valiosos extraídos dos dados.

**Tipos de Armazenamento de Big Data**

* **Data Warehouses:** Armazenam dados históricos estruturados para análise e relatórios.
* **Data Lakes:** Repositórios centralizados para todos os tipos de dados, incluindo dados brutos e processados.
* **NoSQL Databases:** Bancos de dados não relacionais projetados para lidar com grandes volumes de dados não estruturados.
* **Distributed File Systems:** Sistemas de arquivos que distribuem dados por vários servidores para maior capacidade e resiliência.

**Fórmulas**

**Fórmula de volume de dados:**

V = n x s

Onde:

* V é o volume de dados
* n é o número de registros
* s é o tamanho do registro

**Fórmula de velocidade de dados:**

D = V / t

Onde:

* D é a taxa de transferência de dados
* V é o volume de dados
* t é o tempo necessário para transferir os dados

**Fórmula de custo de armazenamento:**

C = P x V

Onde:

* C é o custo de armazenamento
* P é o preço por unidade de armazenamento
* V é o volume de dados

Item do edital: Conceitos de processamento massivo e paralelo

**Conceitos de Processamento Massivo e Paralelo**

**Processamento Massivo**

* Envolve o processamento simultâneo de grandes volumes de dados em um único sistema de computador.
* Utilizado para aplicativos que exigem análise e processamento de dados em larga escala, como ciência de dados e aprendizado de máquina.
* Exemplo: processamento de dados meteorológicos para detecção de padrões climáticos.

**Processamento Paralelo**

* Divide tarefas de computação complexas em várias tarefas menores que são processadas simultaneamente em vários processadores ou núcleos.
* Aumenta a velocidade e a eficiência do processamento, pois as tarefas podem ser divididas e distribuídas entre vários recursos de computação.
* Tipos comuns de processamento paralelo:
  + **Paralelismo entre tarefas:** Cada tarefa é executada simultaneamente em um processador separado.
  + **Paralelismo de dados:** Os mesmos dados são processados por vários processadores, cada um responsável por uma parte diferente dos dados.
  + **Paralelismo de canalização:** As tarefas são organizadas em uma ordem específica, com a saída de uma tarefa servindo como entrada para a próxima.

**Fórmula para Escalabilidade de Paralelismo:**

T(p) = (T(1) + O) / p

* **T(p):** Tempo de execução com "p" processadores
* **T(1):** Tempo de execução com um processador (serial)
* **O:** Sobrecarga da comunicação e sincronização

A lei de Amdahl afirma que a escalabilidade de um programa paralelo é limitada pelo tempo gasto na parte serial do programa.

**Benefícios do Processamento Massivo e Paralelo**

* Aumento da velocidade e eficiência de processamento
* Capacidade de lidar com conjuntos de dados maiores
* Análise e insights mais rápidos e precisos
* Redução do tempo de processamento geral

Item do edital: Processamento distribuído

**Processamento Distribuído**

O processamento distribuído é uma abordagem de computação que divide uma tarefa em subtarefas menores e as executa simultaneamente em vários computadores conectados em rede.

**Benefícios**

* **Paralelismo:** Permite que tarefas sejam executadas simultaneamente, reduzindo o tempo de execução.
* **Escalabilidade:** A capacidade de adicionar ou remover computadores ao sistema conforme necessário, escalando o desempenho.
* **Tolerância a falhas:** Se um computador falhar, outras partes do sistema podem continuar a operar, minimizando o impacto da falha.
* **Eficiência de custos:** Pode ser mais econômico usar vários computadores baratos do que um único computador poderoso.

**Tipos de Arquiteturas de Processamento Distribuído**

* **Cluster:** Um grupo de computadores conectados em rede que compartilham recursos.
* **Grid Computing:** Um conjunto de computadores geograficamente dispersos que são agregados para formar um único recurso computacional.
* **Nuvem:** Um conjunto de recursos de computação que são fornecidos como um serviço por meio da Internet.

**Fórmulas**

* **Lei de Amdahl:** O tempo total de execução de um programa usando processamento distribuído é determinado pela parte sequencial do programa.
* **Lei de Gustafson:** O tempo total de execução de um programa usando processamento distribuído escala linearmente com o número de processadores se o programa for completamente paralelizável.

**Aplicações**

O processamento distribuído é amplamente usado em vários domínios, incluindo:

* Simulações científicas e numéricas
* Processamento de dados grande (Big Data)
* Reconhecimento de padrões e aprendizado de máquina
* Renderização gráfica

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

**Arquitetura do Ecossistema Spark para Soluções de Big Data**

**Introdução:** Apache Spark é uma plataforma de computação distribuída para processamento de big data. Sua arquitetura de ecossistema oferece uma gama de componentes para construir soluções de big data escaláveis e eficientes.

**Componentes principais:**

**1. Spark Core:**

* O núcleo do Spark, fornecendo primitivas de RDD (conjuntos de dados resilientes distribuídos) e APIs para transformações e ações de dados.

**2. Spark SQL:**

* Módulo para processamento analítico de dados estruturados, incluindo suporte a SQL e DataFrames.

**3. Spark Streaming:**

* Para processamento de dados em tempo real, recebendo dados de várias origens.

**4. Spark MLlib:**

* Biblioteca para aprendizado de máquina, fornecendo algoritmos e utilitários para modelamento e previsão.

**5. Spark GraphX:**

* Para processamento gráfico, permitindo operações como travessia e análise de conexões.

**Ferramentas de ecossistema:**

* **Apache Hadoop:** Sistema de arquivos distribuído que fornece armazenamento para dados do Spark.
* **Apache Cassandra:** Banco de dados NoSQL para dados estruturados e semiestruturados.
* **Apache Kafka:** Sistema de mensagens de streaming para processamento de dados em tempo real.
* **Apache Zeppelin:** Bloco de notas da web que permite fácil acesso aos dados e ferramentas do Spark.

**Arquitetura de implantação:**

O Spark pode ser implantado em vários modos, incluindo:

* **Cluster autônomo:** Implante o Spark em nós independentes, gerenciando recursos e coordenação.
* **Modo YARN:** Implante o Spark no cluster Hadoop YARN, aproveitando o gerenciamento de recursos do YARN.
* **Modo Kubernetes:** Implante o Spark no cluster Kubernetes, oferecendo flexibilidade e escalabilidade.

**Benefícios:**

* **Processamento rápido:** O mecanismo de execução com otimização de memória do Spark permite processamento de dados de alta velocidade.
* **Escalabilidade:** A arquitetura distribuída permite escalabilidade linear com a adição de nós.
* **Facilidade de uso:** APIs intuitivas e ferramentas de desenvolvimento simplificam o desenvolvimento de soluções de big data.
* **Suporte a vários formatos de dados:** O Spark pode processar dados de vários formatos, incluindo CSV, JSON, Parquet e ORC.

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

**Arquitetura de Cloud para Ciência de Dados**

**AWS (Amazon Web Services)**

* **Serviços de Computação:** EC2 (instâncias de computação em nuvem), Lambda (funções serverless)
* **Armazenamento de Dados:** S3 (armazenamento de objetos), RDS (bancos de dados relacionais)
* **Processamento de Dados:** Athena (processamento de dados SQL), Redshift (data warehouse)

**Azure (Microsoft Azure)**

* **Serviços de Computação:** Virtual Machines (instâncias de computação em nuvem), Azure Functions (funções serverless)
* **Armazenamento de Dados:** Blob Storage (armazenamento de objetos), SQL Server (bancos de dados relacionais)
* **Processamento de Dados:** Data Lake (armazenamento de dados), Synapse Analytics (plataforma de análise de dados)

**GCP (Google Cloud Platform)**

* **Serviços de Computação:** Compute Engine (instâncias de computação em nuvem), Cloud Functions (funções serverless)
* **Armazenamento de Dados:** Cloud Storage (armazenamento de objetos), BigQuery (data warehouse)
* **Processamento de Dados:** BigQuery ML (aprendizado de máquina), Vertex AI (plataforma de IA)

**Principais Benefícios:**

* **Escalabilidade:** Possibilidade de expandir ou reduzir rapidamente os recursos conforme necessário.
* **Flexibilidade:** Suporte a várias linguagens de programação e ferramentas de ciência de dados.
* **Redução de custos:** Pagamento apenas pelos recursos usados, eliminando custos com infraestrutura própria.
* **Colaboração:** Ferramentas de colaboração integradas facilitam o trabalho em equipe.
* **Inteligência Artificial (IA):** Os serviços de IA integrados aprimoram as capacidades de ciência de dados.

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

**Álgebra Relacional**

A Álgebra Relacional é um conjunto de operações formais para manipular relações (tabelas). É baseada na teoria dos conjuntos e fornece uma notação matemática para expressar consultas de banco de dados.

**Operações da Álgebra Relacional**

* **Seleção (σ):** Seleciona linhas com base em uma condição. **Sintaxe:** σ[condição] (relação)
* **Projeção (π):** Seleciona colunas específicas. **Sintaxe:** π[lista de colunas] (relação)
* **Produto Cartesiano (x):** Combina linhas de duas relações. **Sintaxe:** relação1 x relação2
* **Junção (⋈):** Combina linhas com base em um predicado de junção. **Sintaxe:** relação1 ⋈[predicado de junção] relação2
* **União (∪):** Combina linhas duplicadas removendo linhas repetidas. **Sintaxe:** relação1 ∪ relação2
* **Interseção (∩):** Retorna linhas comuns a ambas as relações. **Sintaxe:** relação1 ∩ relação2
* **Diferença (-):** Retorna linhas na primeira relação que não estão na segunda. **Sintaxe:** relação1 - relação2

**SQL (padrão ANSI)**

SQL (Structured Query Language) é uma linguagem de consulta declarativa usada para interagir com bancos de dados relacionais. Baseia-se na Álgebra Relacional e fornece uma sintaxe amigável para expressar consultas.

**Sintaxe SQL Básica**

* **SELECT:** Seleciona colunas de uma tabela.
* **FROM:** Especifica a tabela(s) envolvida(s).
* **WHERE:** Define a condição de seleção.
* **JOIN:** Combina tabelas com base em predicados de junção.
* **UNION:** Combina linhas de várias consultas.
* **INTERSECT:** Retorna linhas comuns a várias consultas.
* **EXCEPT:** Retorna linhas na primeira consulta que não estão nas outras.

**Conclusão**

A Álgebra Relacional fornece uma base matemática para manipular relações, enquanto SQL é uma linguagem prática para expressar consultas de banco de dados. Ambas são ferramentas poderosas para recuperar e manipular dados em sistemas de banco de dados relacionais.

Item do edital: SQL Server

**SQL Server**

SQL Server é um poderoso sistema gerenciador de banco de dados relacional (SGBDR) desenvolvido pela Microsoft. Ele oferece recursos abrangentes para gerenciamento e análise de dados.

**Recursos Principais:**

* **Armazenamento de Dados:** Armazena e gerencia grandes quantidades de dados em tabelas, linhas e colunas.
* **Linguagem SQL:** Usa a linguagem SQL (Structured Query Language) para criar, manipular e consultar dados.
* **Indexação:** Cria e gerencia índices para acelerar as consultas.
* **Transações:** Garante a integridade dos dados durante as atualizações.
* **Backup e Recuperação:** Fornece mecanismos robustos para backup e recuperação de dados.
* **Ferramentas de Inteligência de Negócios:** Inclui ferramentas analíticas integradas para explorar e visualizar dados.
* **Alta Disponibilidade:** Suporta recursos de alta disponibilidade para garantir continuidade dos negócios.

**Fórmulas:**

* **Fórmula de Armazenamento de Dados:**
  + Tabela = Coleção de linhas
  + Linha = Coleção de colunas
  + Coluna = Atributo que armazena valores específicos
* **Fórmula de Indexação:**
  + Índice = Estrutura de dados que mapeia valores de coluna para endereços de linha
  + Melhor desempenho da consulta devido à pesquisa mais rápida por valores específicos
* **Fórmula de Transação:**
  + Transação = Unidade lógica de trabalho
  + ACID (Atomicidade, Consistência, Isolamento, Durabilidade): Garante a integridade dos dados mesmo em caso de falhas

**Benefícios:**

* Alta desempenho
* Escalabilidade
* Segurança robusta
* Gerenciamento de dados eficiente
* Ferramentas analíticas integradas

**Aplicações:**

* Sistemas de gerenciamento de clientes (CRM)
* Sistemas de planejamento de recursos empresariais (ERP)
* Bancários e financeiros
* Saúde
* Varejo

Item do edital: PostgreSQL

**PostgreSQL (PostGRES)**

**Conceito:**

PostgreSQL é um Sistema Gerenciador de Banco de Dados (SGBD) relacional de código aberto e objeto-relacional que destaca pela confiabilidade, escalabilidade e recursos avançados.

**Características:**

* **Transações Ácidas:** Suporta transações ACID (atomicidade, consistência, isolamento, durabilidade).
* **Suporte a Tipos de Dados Avançados:** Oferece uma ampla gama de tipos de dados, incluindo arrays, JSON, geográficos, temporais e monetários.
* **Índices Personalizados:** Permite a criação de índices personalizados para otimizar consultas.
* **Particionamento:** Permite dividir bancos de dados grandes em partes menores para melhorar o desempenho.
* **Replicação:** Suporta replicação síncrona e assíncrona para alta disponibilidade e tolerância a falhas.

**Fórmulas:**

A fórmula para calcular o uso de memória do PostgreSQL é:

Uso de Memória (MB) = Shared Buffers + Cache de Conexões + Cachê de Consulta

Onde:

* **Shared Buffers:** Buffers compartilhados por todos os processos.
* **Cache de Conexões:** Cache de conexões de clientes.
* **Cache de Consulta:** Cache de consultas para melhorar o desempenho.

**Vantagens:**

* Código aberto e gratuito.
* Confiável e escalável.
* Recursos avançados para necessidades complexas de dados.
* Grande comunidade de suporte e documentação.

**Aplicações:**

PostgreSQL é amplamente utilizado para:

* Armazenamento e gerenciamento de grandes conjuntos de dados.
* Sistemas de gerenciamento de informações geográficas (GIS).
* Aplicativos web e móveis.
* Sistemas de data warehouse.

Item do edital: MySQL

**MySQL**

O MySQL é um sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto e gratuito, amplamente utilizado para gerenciar dados em aplicativos da web, software e sistemas corporativos.

**Principais recursos:**

* **Modelo de dados relacional:** Organiza dados em tabelas relacionadas por meio de chaves estrangeiras.
* **Linguagem de consulta estruturada (SQL):** Permite acesso e manipulação de dados usando comandos SQL.
* **Transações ACID:** Garante a integridade dos dados ao executar operações em conjuntos atômicos.
* **Otimizações de desempenho:** Inclui recursos como índices, caches e paralelismo para aprimorar o desempenho das consultas.
* **Escalabilidade:** Suporta cargas de trabalho de grande escala com replicação e particionamento.
* **Alta disponibilidade:** Opções de replicação e clustering garantem a disponibilidade ininterrupta de dados.

**Termos-chave:**

* **Banco de dados:** Coleção de tabelas que armazenam dados relacionados.
* **Tabela:** Estrutura que contém linhas (registros) e colunas (atributos).
* **Chave primária:** Identificador exclusivo para cada linha em uma tabela.
* **Chave estrangeira:** Coluna que referencia uma chave primária em outra tabela, vinculando-as.
* **Índices:** Estruturas de dados que aceleram a pesquisa por valores comuns.
* **Consulta:** Comando SQL usado para recuperar ou modificar dados.
* **Transação:** Conjunto de operações que são executadas como uma única unidade.

**Fórmulas:**

**Índice de cobertura:**

Índice de Cobertura = (Número de linhas acessadas pelo índice) / (Número total de linhas na tabela)

**Taxa de acerto do cache:**

Taxa de Acerto do Cache = (Número de acessos ao cache bem-sucedidos) / (Número total de acessos ao cache)

Item do edital: Bnco de dados NoSQL.

**Bancos de Dados NoSQL**

**Definição:**

Bancos de Dados NoSQL (Não Relacionais) são sistemas de gerenciamento de dados que não aderem ao modelo relacional tradicional. Eles são projetados para lidar com dados massivos, estruturados ou não estruturados.

**Características Principais:**

* **Flexibilidade do Esquema:** Sem schema rígido, permitindo dados de vários formatos e estruturas.
* **Escala Horizontal:** Capacidade de dividir e distribuir dados em vários servidores para maior escalabilidade.
* **Modelo de Dados Não Relacional:** Usa modelos de dados como chave-valor, documentos, colunas ou grafos.
* **Consistência Eventual:** Geralmente oferecem consistência eventual em vez de consistência imediata.
* **Alta Disponibilidade:** Projetados para minimizar o tempo de inatividade e manter a disponibilidade dos dados.

**Tipos de Bancos de Dados NoSQL:**

* **Chave-Valor:** Armazena pares chave-valor simples (por exemplo, Redis, DynamoDB)
* **Documentos:** Armazena dados como documentos JSON ou XML (por exemplo, MongoDB, CouchDB)
* **Colunas:** Divide os dados em colunas em vez de linhas (por exemplo, Cassandra, HBase)
* **Grafos:** Representa dados como um grafo, onde os nós são entidades e as arestas são relacionamentos (por exemplo, Neo4j, OrientDB)

**Fórmulas:**

**Tempo Médio de Reparo (MTTR):**

MTTR = Tempo de Detecção do Problema + Tempo de Isolamento do Problema + Tempo de Recuperação do Problema

**Tempo Médio Entre Falhas (MTBF):**

MTBF = Tempo Total de Operação / Número de Falhas

**Vantagens:**

* Alta escalabilidade
* Flexibilidade de dados
* Alta disponibilidade
* Custos mais baixos para dados massivos

**Desvantagens:**

* Consistência eventual (pode haver atrasos nas atualizações)
* Complexidade potencial para gerenciar dados complexos
* Suporte limitado a consultas relacionais tradicionais

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

**Banco de Dados Orientado a Colunas**

Um banco de dados orientado a colunas (CDBMS) armazena e gerencia dados organizados verticalmente em colunas, diferentemente dos bancos de dados orientados a linhas, que armazenam dados em linhas.

**Características:**

* **Estrutura de Dados:** Os dados são divididos em colunas, cada uma contendo um tipo de dado específico.
* **Compactação:** Colunas contendo valores semelhantes são compactadas, economizando espaço de armazenamento.
* **Consulta Otimizada:** As consultas podem ser processadas mais rapidamente, pois os dados relevantes são agrupados em colunas.

**Formatos de Arquivo Orientados a Colunas**

* **Apache Parquet:** Formato de arquivo binário aberto projetado para armazenar grandes conjuntos de dados.
* **Apache ORC:** Formato de arquivo de código aberto otimizado para processamento rápido de consultas.
* **RCFile:** Formato de arquivo proprietário do Apache Hadoop, projetado para eficiência de armazenamento.

**Fórmulas:**

* **Tempo de Consulta:** Para um determinado conjunto de dados:
  + CDBMS orientado a linhas: T = N \* k \* log(n)
  + CDBMS orientado a colunas: T = k \* log(n)
  + Onde:
    - T é o tempo de consulta
    - N é o número de linhas
    - k é o número de colunas selecionadas
    - n é o número total de colunas

**Vantagens:**

* Eficiência de armazenamento para conjuntos de dados esparsos
* Consultas mais rápidas, especialmente para consultas de agregação
* Escalabilidade para grandes conjuntos de dados

**Desvantagens:**

* Pode ser menos eficiente para consultas que requerem junções ou atualizações frequentes
* Pode ter maior complexidade para desenvolvimento e manutenção do esquema

Item do edital: Banco de dados Parquet

**Banco de Dados Parquet**

Parquet é um formato de arquivo de código aberto projetado para armazenar dados em colunas. É otimizado para processamento analítico de grandes conjuntos de dados e oferece vários recursos e benefícios:

**Características:**

* **Armazenamento em Colunas:** Armazena dados em colunas, o que melhora a eficiência de leitura e gravação para consultas de colunas específicas.
* **Compressão:** Suporta vários algoritmos de compressão, como GZIP, Snappy e LZO, para reduzir o tamanho do arquivo.
* **Esquema Definido:** Define um esquema rígido, garantindo a consistência dos dados e facilitando o processamento.
* **Suporte a Partições:** Permite dividir conjuntos de dados em partes menores para gerenciamento e processamento eficientes.
* **Índices:** Suporta a criação de índices em colunas para melhorar o desempenho de pesquisa.

**Vantagens:**

* **Eficiência de Leitura:** A estrutura em colunas permite leitura eficiente de subconjuntos de dados, reduzindo o tempo de consulta.
* **Compressão Eficaz:** A compressão incorporada reduz o tamanho do arquivo, economizando espaço de armazenamento e largura de banda.
* **Escala:** Projetado para lidar com conjuntos de dados muito grandes, escalando bem em clusters distribuídos.
* **Processamento Paralelo:** O suporte nativo a partições e índices facilita o processamento paralelo de dados.
* **Integração Ampla:** Compatível com vários frameworks analíticos, como Spark, Hadoop e Hive.

**Fórmulas de Compressão:**

Os algoritmos de compressão usados pelo Parquet incluem:

* **Deflate (GZIP):** (1 - (Tamanho do arquivo comprimido / Tamanho do arquivo original)) \* 100%
* **Snappy:** (1 - (Tamanho do arquivo comprimido / Tamanho do arquivo original)) \* 100%
* **LZO:** (1 - (Tamanho do arquivo comprimido / Tamanho do arquivo original)) \* 100%

Item do edital: MonetDB

**MonetDB**

MonetDB é um Sistema de Gerenciamento de Banco de Dados Relacional de Código Aberto (RDBMS), otimizado para processamento analítico online (OLAP), processamento de dados em memória e escala massiva.

**Características Principais:**

* **Colunar:** Armazena dados em colunas, permitindo acesso rápido e eficiente a conjuntos de dados específicos.
* **Memória Principal:** Mantém dados em memória principal (RAM), proporcionando desempenho excepcionalmente rápido.
* **Escala Massiva:** Suporta cargas de trabalho enormes, gerenciando trilhões de linhas em um único nó.
* **MPP (Processamento Massivo Paralelo):** Divide consultas em subtarefas menores que são executadas em paralelo em vários nós, aumentando o desempenho.
* **Linguagem de Consulta Personalizada (MQL):** Oferece uma linguagem de consulta poderosa e concisa, otimizada para análise de dados.

**Formula:**

O MonetDB é construído sobre o Modelo de Dados Colunar, onde os dados são armazenados em colunas em vez de linhas. Esta organização de dados permite acesso rápido e eficiente a conjuntos de dados específicos, pois apenas as colunas necessárias são lidas da memória.

**Benefícios:**

* Excelentes tempos de resposta para consultas complexas
* Manipulação eficiente de grandes conjuntos de dados
* Escalabilidade horizontal para cargas de trabalho massivas
* Custo-benefício em comparação com outros RDBMS
* Ideal para análise de dados, data warehousing e inteligência de negócios

**Casos de Uso:**

* Análise de Dados em Tempo Real
* Processamento de Grandes Volumes de Dados
* Business Intelligence e Relatórios
* Gerenciamento de Dados Financeiros
* Sistemas de Recomendação

Item do edital: duckDB.

**DuckDB**

DuckDB é um banco de dados relacional analítico de código aberto projetado para processamento rápido de dados analíticos em memória.

**Características:**

* **Colunar:** Armazena dados em colunas, otimizando consultas de leitura.
* **Em memória:** Carrega dados inteiramente na memória, permitindo operações de consulta rápidas.
* **SQL Compatível:** Suporta a linguagem SQL padrão, tornando fácil migrar de outros bancos de dados.
* **Extensível:** Possui uma API que permite que os usuários criem funções e operadores personalizados.

**Fórmulas:**

**Criação de tabela:**

CREATE TABLE tabela\_nome (

coluna1 tipo\_dados,

coluna2 tipo\_dados

);

**Inserção de dados:**

INSERT INTO tabela\_nome (coluna1, coluna2)

VALUES (valor1, valor2);

**Consulta de dados:**

SELECT \* FROM tabela\_nome

WHERE coluna1 > valor;

**Agregação de dados:**

SELECT SUM(coluna1) FROM tabela\_nome

GROUP BY coluna2;

**Vantagens:**

* **Desempenho rápido:** Consultas são processadas em milissegundos, mesmo em conjuntos de dados grandes.
* **Economia de memória:** A arquitetura colunar minimiza o uso de memória.
* **Facilidade de uso:** A interface SQL familiar facilita o uso para analistas de dados.
* **Portabilidade:** Disponível em vários sistemas operacionais, incluindo Windows, Linux e macOS.

**Limitações:**

* **Capacidade de armazenamento:** Limitado por memória disponível.
* **Durabilidade:** Não garante persistência de dados após reinicializações.
* **Transações:** Não suporta transações tradicionais, mas fornece mecanismos alternativos de controle de simultaneidade.

Item do edital: Normalização numérica.

**Normalização Numérica**

A normalização numérica é o processo de padronizar os valores numéricos para ficarem dentro de uma faixa específica. Isso é feito para melhorar a precisão e a estabilidade dos algoritmos numéricos.

**Motivações**

* **Escala de valores:** Números muito grandes ou muito pequenos podem causar estouro ou subfluxo em operações aritméticas.
* **Precisão:** Números muito diferentes em magnitude podem introduzir erros de arredondamento significativos.
* **Estabilidade:** Operações como divisão e subtração podem se tornar instáveis se os números tiverem magnitudes muito diferentes.

**Métodos de Normalização**

**Normalização Linear:**

* Normaliza os valores para estarem entre 0 e 1, ou -1 e 1:

x\_norm = (x - x\_min) / (x\_max - x\_min)

onde x\_min e x\_max são os valores mínimo e máximo do conjunto de dados original.

**Normalização de Desvio Padrão:**

* Normaliza os valores para terem média 0 e desvio padrão 1:

x\_norm = (x - μ) / σ

onde μ é a média e σ é o desvio padrão do conjunto de dados original.

**Normalização Min-Max:**

* Normaliza os valores para estarem entre o valor mínimo e máximo possíveis:

x\_norm = (x - x\_min) / (x\_max - x\_min)

**Normalização Logarítmica:**

* Normaliza os valores tomando seus logaritmos:

x\_norm = log(x)

**Vantagens**

* Melhora a precisão e a estabilidade dos cálculos.
* Torna os algoritmos mais robustos a erros de arredondamento.
* Facilita a comparação de valores de magnitudes diferentes.

**Desvantagens**

* Pode alterar a distribuição dos dados.
* Pode introduzir vieses se os dados não forem distribuídos normalmente.

Item do edital: Discretização.

**Discretização**

Discretização é o processo de converter um sinal ou processo contínuo em um sinal ou processo discreto. Isso envolve amostrar o sinal contínuo em intervalos regulares e representar seus valores em um formato discreto, geralmente como números ou valores binários.

**Métodos de Discretização**

Existem vários métodos de discretização, incluindo:

* **Amostragem Uniform:** Amostragem em intervalos de tempo ou espaço uniformes.
* **Amostragem Adaptativa:** Amostragem com base na taxa de variação do sinal.
* **Amostragem Aleatória:** Amostragem em intervalos de tempo ou espaço aleatórios.

**Fórmula de Amostragem**

A fórmula geral para discretizar um sinal contínuo f(t) é:

x[n] = f(nT)

onde:

* x[n] é o sinal discretizado
* f(t) é o sinal contínuo
* T é o intervalo de amostragem

**Limite de Nyquist**

O Teorema da Amostragem de Nyquist afirma que a taxa de amostragem deve ser pelo menos duas vezes a largura de banda do sinal contínuo para evitar o aliasing, que é a introdução de distorções no sinal discretizado.

**Vantagens da Discretização**

* Permite o processamento digital de sinais usando computadores.
* Reduz a quantidade de dados necessária para representar um sinal.
* Facilita a transmissão e o armazenamento de sinais.

**Desvantagens da Discretização**

* Pode causar perda de informação se a taxa de amostragem for muito baixa.
* Introduz latência devido ao tempo necessário para amostrar e processar o sinal.
* Pode ser sensível a ruído e interferências.

Item do edital: Tratamento de dados ausentes.

**Tratamento de Dados Ausentes**

O tratamento de dados ausentes é uma tarefa crucial na análise de dados, pois os dados ausentes podem comprometer a precisão e a validade dos resultados. Existem várias técnicas para lidar com dados ausentes, incluindo:

**Remoção de Casos**

* **Exclusão por lista:** Remove casos com dados ausentes específicos.
* **Remoção completa de casos:** Remove todos os casos com qualquer tipo de dado ausente.

**Imputação de Dados**

* **Imputação aleatória:** Substitui dados ausentes por valores gerados aleatoriamente.
* **Imputação por média:** Substitui dados ausentes pelo valor médio da variável.
* **Imputação por vizinhos mais próximos:** Substitui dados ausentes pelos valores dos casos mais próximos no espaço de variáveis.
* **Imputação bayesiana:** Substitui dados ausentes por valores amostrados a partir de uma distribuição posterior condicional.

**Análise de Múltiplas Imputações (MIA)**

* Imputa os dados ausentes várias vezes, cria vários conjuntos de dados completos e analisa cada conjunto separadamente.
* Combina os resultados das análises individuais para obter estimativas imputadas.

**Fórmulas**

**Imputação por média:**

Valor imputado = Média da variável

**Imputação por vizinhos mais próximos:**

Valor imputado = Valor do caso mais próximo no espaço de variáveis

**Análise de Múltiplas Imputações:**

Estimativa imputada = Média das estimativas imputadas + √(Entre + Dentro) / nimp

onde:

* Entre = Variância entre as estimativas imputadas
* Dentro = Variância dentro das estimativas imputadas
* nimp = Número de imputações

**Considerações na Escolha da Técnica**

A escolha da técnica de tratamento de dados ausentes depende de fatores como:

* Taxa e padrão de dados ausentes
* Tipo de dados (numéricos ou categóricos)
* Distribuição da variável
* Objetivos da análise

Item do edital: Tratamento de outliers e agregações.

**Tratamento de Outliers**

Outliers são dados que são significativamente diferentes do resto dos dados em um conjunto. Eles podem distorcer os resultados da análise e levar a conclusões incorretas. Existem várias técnicas para tratar outliers:

* **Exclusão:** Remover os outliers do conjunto de dados.
* **Substituição:** Substituir os outliers por valores estimados, como a mediana ou a média do restante do conjunto de dados.
* **Transformação:** Transformar os dados para reduzir o impacto dos outliers, como usando uma transformação logarítmica ou quadrada.
* **Winsorização:** Substituir os valores extremos dos outliers pelos valores do percentil mais alto ou mais baixo do restante do conjunto de dados.

**Agregações**

Agregações são funções que resumem os dados em um único valor. Elas são usadas para simplificar a análise e identificar tendências. As agregações mais comuns incluem:

* **Soma:** Adiciona todos os valores no conjunto de dados.
* **Média:** Calcula a soma dos valores e divide pelo número de valores.
* **Mediana:** O valor do meio quando os dados são ordenados.
* **Moda:** O valor que ocorre com mais frequência.
* **Desvio padrão:** Uma medida da dispersão dos dados em torno da média.
* **Covariância:** Uma medida da relação linear entre duas variáveis.
* **Correlação:** Uma medida da força e direção da relação linear entre duas variáveis.

**Fórmulas de Agregação**

* Média: x̄ = (Σx) / n
* Mediana: Ordenar os dados do menor para o maior e selecionar o valor do meio.
* Moda: O valor que ocorre com mais frequência.
* Desvio padrão: σ = √[(Σ(x - x̄)²) / (n - 1)]
* Covariância: cov(x, y) = (Σ(x - x̄)(y - ȳ)) / (n - 1)
* Correlação: corr(x, y) = cov(x, y) / (σxσy)

Item do edital: Tratamento de dados: Matching

**Tratamento de Dados: Matching**

O matching é uma técnica de tratamento de dados que consiste em combinar registros de dados de duas ou mais fontes diferentes com base em critérios pré-definidos, como uma chave comum ou características semelhantes.

**Objetivos do Matching:**

* Remover duplicatas
* Enriquecer dados com informações de fontes adicionais
* Identificar relacionamentos entre dados diferentes

**Métodos de Matching:**

* **Matching Determinista:** Usa uma chave comum (por exemplo, ID do cliente) para combinar registros com precisão.
* **Matching Probabilístico:** Usa algoritmos para calcular a probabilidade de dois registros serem correspondentes com base em características comuns (por exemplo, nome, endereço).

**Fórmulas de Matching Probabilístico:**

* **Coeficiente de Jaccard:** mede a similaridade entre dois conjuntos de atributos.
* **Coeficiente de Dice:** semelhante ao coeficiente de Jaccard, mas pondera os atributos em comum.
* **Distância de Levenshtein:** mede o número de operações de edição (inserção, remoção, substituição) necessárias para transformar uma string em outra.

**Etapas do Matching:**

1. Definição dos critérios de correspondência
2. Preparação dos dados (por exemplo, padronização, remoção de dados ausentes)
3. Aplicação do algoritmo de matching
4. Avaliação da precisão e cobertura do matching
5. Resolução de quaisquer correspondências incertas

**Benefícios do Matching:**

* Melhora a qualidade dos dados
* Reduz custos ao eliminar dados duplicados
* Permite análises de dados mais precisas
* Facilita a tomada de decisões baseada em informações mais completas

Item do edital: Deduplicação.

**Dedução**

A deduplicação é um processo de identificação e remoção de cópias duplicadas de dados para reduzir o espaço de armazenamento e melhorar a eficiência. É comumente usado em sistemas de armazenamento de dados, backup e arquivamento.

**Princípios**

* **Identificação de Duplicatas:** Algoritmos de hash ou comparações de dados são usados para identificar blocos de dados duplicados.
* **Armazenamento Referencial:** Uma única cópia do bloco deduzido é armazenada, enquanto todas as outras referências apontam para a cópia armazenada.

**Métodos**

Existem dois métodos principais de deduplicação:

* **Dedução Inline:** Ocorre durante a gravação de dados, identificando e removendo duplicatas antes do armazenamento.
* **Dedução Pós-Processo:** Ocorre após os dados terem sido armazenados, identificando e removendo duplicatas posteriormente.

**Vantagens**

* **Redução de Armazenamento:** Pode reduzir significativamente o espaço de armazenamento necessário.
* **Melhoria de Desempenho:** Menos dados precisam ser processados, levando à melhoria do desempenho do sistema.
* **Redução de Custo:** A redução do espaço de armazenamento leva a custos mais baixos de armazenamento.

**Desvantagens**

* **Sobrecarga de Processamento:** O processo de deduplicação pode consumir recursos de processamento.
* **Recuperação de Dados Mais Complexa:** Os dados deduzidos são armazenados de forma referenciada, tornando a recuperação de dados mais complexa.

**Fórmulas**

A taxa de deduplicação é calculada como:

Taxa de Deduplicação = Tamanho dos Dados Deduplicados / Tamanho dos Dados Originais

Por exemplo, se os dados originais têm 100 GB e os dados deduzidos têm 50 GB, a taxa de deduplicação é de 2:1.

Item do edital: Data cleansing.

**Data Cleansing**

Data cleansing é o processo de identificar e corrigir inconsistências, erros e dados ausentes em um conjunto de dados para torná-lo mais preciso e confiável.

**Etapas:**

1. **Coleta de dados:** Coletar dados de várias fontes para identificar lacunas e erros.
2. **Validação:** Verificar se os dados atendem a critérios específicos, como formato, intervalo de valores etc.
3. **Transformação:** Converter ou padronizar dados para garantir consistência.
4. **Imputação:** Preencher dados ausentes usando técnicas como:
   * **Imputação de média:** Preencher valores ausentes com a média dos valores não ausentes.
   * **Imputação de mediana:** Preencher valores ausentes com a mediana dos valores não ausentes.
   * **Imputação de K vizinhos mais próximos (KNN):** Preencher valores ausentes com a média dos valores dos K vizinhos mais próximos no espaço de recursos.
5. **Detecção de outliers:** Identificar valores extremamente altos ou baixos que podem distorcer a análise.
6. **Remoção de duplicatas:** Remover registros duplicados para garantir a integridade dos dados.

**Fórmulas:**

* **Média:** (x1 + x2 + ... + xn) / n
* **Mediana:** O valor médio no conjunto de dados classificado.
* **KNN:** (y1 + y2 + ... + yk) / k

**Benefícios:**

* Melhora a precisão da análise de dados
* Reduz erros e inconsistências
* Aumenta a confiabilidade dos insights
* Facilita a tomada de decisão informada
* Permite que os modelos de aprendizado de máquina sejam mais eficazes

Item do edital: Enriquecimento de dados.

**Enriquecimento de Dados**

O enriquecimento de dados é o processo de aprimorar os dados existentes adicionando informações adicionais relevantes de fontes internas ou externas. Ele visa melhorar a qualidade, a integridade e a relevância dos dados para fins de tomada de decisão, análise e insights mais precisos.

**Etapas do Enriquecimento de Dados:**

1. **Coleta de Dados:** Identificação e coleta de fontes de dados internas e externas.
2. **Limpeza e Preparação:** Remoção de duplicatas, correção de erros e formatação dos dados para análise.
3. **Enriquecimento Real:** Fusão de dados novos com dados existentes usando técnicas como:
   * **Junção por Chave:** Vinculação de registros com base em uma chave comum (por exemplo, ID do cliente).
   * **Enriquecimento por Probabilidade:** Correspondência de registros com base em atributos semelhantes, com nível de confiança atribuído.
   * **Enriquecimento Hierárquico:** Adição de dados hierárquicos, como categorias ou taxonomias.
4. **Validação e Verificação:** Avaliação da precisão e integridade dos dados enriquecidos para garantir confiabilidade.

**Fórmulas:**

**Fórmula para Pontuação de Correspondência:**

Pontuação = Σ(Peso x Semelhança)

Onde:

* Peso é o peso atribuído a cada atributo de correspondência.
* Semelhança é o grau de similaridade entre os valores do atributo.

**Fórmula para Normalização de Dados:**

Valor Normalizado = (Valor - Valor Mínimo) / (Valor Máximo - Valor Mínimo)

**Benefícios do Enriquecimento de Dados:**

* Melhor tomada de decisão com base em insights mais precisos.
* Maior eficiência operacional e redução de custos.
* Experiências personalizadas do cliente e marketing direcionado.
* Detecção e prevenção de fraudes.
* Inovação e desenvolvimento de novos produtos ou serviços.

Item do edital: Desidentificação de dados sensíveis.

**Desidentificação de Dados Sensíveis**

A desidentificação de dados é o processo de remover ou mascarar informações de identificação pessoal (PII) de um conjunto de dados, mantendo sua utilidade para análise. O objetivo é proteger a privacidade dos indivíduos enquanto permite o uso dos dados para pesquisas, análises e desenvolvimento de produtos.

**Tipos de Desidentificação**

Existem dois tipos principais de desidentificação:

* **Desidentificação Estática:** Remove PII permanentemente do conjunto de dados.
* **Desidentificação Dinâmica:** Mascara ou oculta PII temporariamente, permitindo acesso autorizado a dados protegidos.

**Técnicas de Desidentificação**

Os métodos específicos usados para desidentificar dados variam dependendo do tipo de dados e do nível desejado de proteção da privacidade. Algumas técnicas comuns incluem:

* **Mascaramento:** Substituir PII por valores fictícios ou aleatórios.
* **Pseudonimização:** Substituir PII por identificadores exclusivos que não podem ser vinculados a indivíduos específicos.
* **Perturbação:** Adicionar ruído ou variação a dados para obscurecer informações confidenciais.
* **Generalização:** Agrupar ou agregar dados para reduzir a especificidade que pode identificar indivíduos.

**Fórmulas**

A eficácia da desidentificação pode ser medida usando a métrica de risco de reidentificação (RR), que estima a probabilidade de um indivíduo ser reidentificado em um conjunto de dados desidentificado.

**RR = P(R|D,A)**

Onde:

* RR é o risco de reidentificação
* P(R|D,A) é a probabilidade de reidentificação dado o conjunto de dados desidentificado (D) e o ataque (A)

**Conclusão**

A desidentificação de dados sensíveis é crucial para proteger a privacidade dos indivíduos e permitir o uso ético de dados para análise e pesquisa. Ao implementar técnicas de desidentificação eficazes, as organizações podem garantir a conformidade com os regulamentos de proteção de dados e manter a confiança dos indivíduos em relação ao uso de seus dados.

Item do edital: Algoritmos fuzzy matching

**Algoritmos Fuzzy Matching**

O Algoritmo Fuzzy Matching é uma técnica usada para encontrar correspondências imprecisas entre conjuntos de dados. Diferentemente de algoritmos de correspondência exata, que requerem correspondências perfeitas, os algoritmos fuzzy matching levam em consideração a similaridade parcial.

**Como Funciona**

Os algoritmos fuzzy matching geralmente calculam uma pontuação de similaridade entre dois registros, que pode variar de 0 (sem similaridade) a 1 (correspondência perfeita). A pontuação é calculada com base na comparação de diferentes atributos, como nomes, endereços e números de telefone.

**Fórmulas Comuns**

* **Coeficiente de Dice:**

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

Onde A e B são conjuntos de atributos.

* **Coeficiente de Jaccard:**

Jaccard(A, B) = |A ∩ B| / |A ∪ B|

Onde A e B são conjuntos de atributos.

* **Distância de Levenshtein:**

Levenshtein(A, B) = mínimo número de inserções, exclusões e substituições necessárias para transformar A em B

**Aplicações**

Os algoritmos fuzzy matching têm diversas aplicações, incluindo:

* Depuração de dados: identificação de registros duplicados e inconsistentes.
* Agregação de dados: combinação de dados de diferentes fontes com atributos parcialmente correspondentes.
* Pesquisa de semelhança: encontrar registros semelhantes com base em atributos de entrada.
* Classificação de texto: agrupar documentos com base em similaridades no conteúdo do texto.

**Limitações**

Embora eficientes, os algoritmos fuzzy matching podem ter limitações:

* Eles podem ser sensíveis à erros de ortografia e variações nas representações de dados.
* A seleção do algoritmo e do limiar de similaridade apropriados pode ser crucial.
* Eles podem ser computacionalmente caros para grandes conjuntos de dados.

Item do edital: Algoritmos stemming.

**Algoritmos Stemming**

Os algoritmos de extração de radicais são técnicas que removem sufixos de palavras (radicais) para obter formas básicas (radicais) que representam seu significado fundamental. Eles são usados em processamento de língua natural para:

* Melhorar o desempenho de motores de busca
* Diminuir a dimensionalidade dos dados de texto
* Remover ruído de dados de texto

**Como funcionam os algoritmos de extração de radicais:**

1. **Dividir a palavra:** A palavra é dividida em sufixos e prefixos.
2. **Remover sufixos:** Sufixos comuns, como "-tion" e "-ity", são removidas da palavra.
3. **Verificar o radical:** O radical resultante é verificado com um dicionário para garantir que seja uma forma válida.

**Tipos comuns de algoritmos de extração de radicais:**

* **Algoritmo de extração de radicais de Martinho:** Um dos algoritmos mais comuns, que usa uma lista exaustiva de regras de extração de radicais.
* **Algoritmo de extração de radicais de Lovin:** Um algoritmo mais sofisticado que leva em conta o contexto das palavras.

**Fórmulas:**

As fórmulas variam entre os algoritmos de extração de radicais específicos, mas aqui está uma fórmula geral para o algoritmo de extração de radicais de Martinho:

radical = palavra.replace(sufixo, "")

**Vantagem:**

* Reduzem o tamanho do vocabulário, melhorando a eficiência computacional.
* Melhoram a recuperação de informações ao expandir as consultas de pesquisa com radicais.

**Desvantagem:**

* Podem levar à ​​remoção excessiva de radicais, resultando em radicais ambíguos.
* Não podem capturar nuances de significado presentes em sufixos removíveis.

Item do edital: Visualização e análise exploratória de dados.

**Visualização de Dados**

A visualização de dados é o processo de representar informações em uma forma gráfica para facilitar a interpretação e compreensão. Técnicas comuns de visualização incluem:

* Gráficos de barras
* Histogramas
* Gráficos de pizza
* Gráficos de dispersão
* Mapas de calor

**Análise Exploratória de Dados (EDA)**

A EDA é um conjunto de técnicas usadas para analisar, explorar e resumir dados sem fazer suposições prévias. O objetivo da EDA é:

* Identificar padrões e tendências
* Detectar valores atípicos
* Testar hipóteses
* Geração de insights acionáveis

**Fórmulas Comuns Usadas na EDA**

* **Média:** Soma dos valores dividida pelo número de valores
* **Mediana:** Valor que divide os dados ao meio
* **Moda:** Valor que ocorre com mais frequência
* **Variância:** Medida da dispersão em relação à média
* **Desvio padrão:** Raiz quadrada da variância

**Etapas da EDA**

* **Limpeza de dados:** Remover valores ausentes, duplicados e erros.
* **Exploração univariada:** Analisar cada variável individualmente usando gráficos e estatísticas.
* **Exploração bivariada:** Analisar o relacionamento entre duas variáveis usando gráficos de dispersão e outras técnicas.
* **Exploração multivariada:** Analisar o relacionamento entre várias variáveis usando técnicas estatísticas mais avançadas.

**Benefícios da Visualização de Dados e EDA**

* Melhor compreensão dos dados
* Identificação de insights acionáveis
* Tomada de decisão mais informada
* Comunicação de informações complexas de forma clara

Item do edital: Linguagem de programação R.

**Linguagem de Programação R**

R é uma linguagem de programação de código aberto amplamente utilizada para análise estatística, aprendizado de máquina e visualização de dados. Ela foi desenvolvida por estatísticos e possui uma vasta gama de recursos para manipulação, análise e apresentação de dados.

**Características:**

* Código aberto e gratuito
* Sintaxe focada em dados, permitindo fácil manipulação de vetores, matrizes e data frames
* Ampla biblioteca de pacotes para tarefas estatísticas, de aprendizado de máquina e de visualização
* Ambiente de desenvolvimento integrado (IDE), conhecido como RStudio, que oferece recursos como preenchimento automático de código e depuração

**Fórmulas:**

* **Modelo de regressão linear:** lm(y ~ x)
* **Distribuição normal:** rnorm(n, mean, sd)
* **Distribuição t de Student:** rt(n, df)

**Vantagens:**

* **Forte orientação estatística:** R foi projetada especificamente para tarefas estatísticas, tornando-a uma escolha ideal para analistas de dados e estatísticos.
* **Grande comunidade:** R possui uma comunidade ativa de usuários e desenvolvedores, fornecendo amplo suporte e recursos.
* **Flexibilidade:** R permite a personalização de funções e algoritmos, permitindo análises complexas.

**Aplicações:**

* **Análise de dados:** Explorar, limpar e analisar dados usando ferramentas estatísticas.
* **Aprendizado de máquina:** Treinar e avaliar modelos de aprendizado de máquina, como árvores de decisão e redes neurais.
* **Visualização de dados:** Criar gráficos e visualizações interativas para apresentar resultados de análise.
* **Desenvolvimento de aplicativos:** Criar aplicativos autônomos usando pacotes como Shiny e Rcpp.

Item do edital: Linguagem de programação Python.

**Linguagem de Programação Python**

**Características Gerais:**

Python é uma linguagem de propósito geral, interpretada, orientada a objetos e multiparadigma. É amplamente reconhecida por sua legibilidade, simplicidade e ampla gama de bibliotecas e frameworks.

**Sintaxe:**

A sintaxe do Python destaca-se por sua simplicidade e claiſter. Ela usa indentação para estrutura de blocos de código, em vez de parênteses ou colchetes. Isso resulta em um código mais limpo e fácil de ler.

**Orientação a Objetos:**

Python é uma linguagem orientada a objetos, o que significa que os dados são representados por objetos que contêm dados e métodos que operam nesses dados. Essa abordagem promove o encapsulamento, herança e polimorfismo.

**Tipos de Variáveis Dinâmicas:**

Python suporta tipagem dinâmicas, o que significa que o tipo de uma variável não é especificado explicitamente durante a declaração. O tipo da variável é determinado dinamicamente durante a execução.

**Biblioteca Padrão Extensa:**

Python possui uma biblioteca padrão abrangente que fornece uma ampla gama de funcionalidades, como processamento de dados, acesso a banco de dados, manipulação de strings e expressão regular. Além disso, há um ecossistema vibrante de bibliotecas e frameworks de terceiros que estendem ainda mais as capacidades do Python.

**Aplicações:**

Python é usado em uma ampla gama de aplicações, que incluem:

* Desenvolvimento Web
* Análise de dados
* Ciência da computação
* Inteligência artificial
* Automação de tarefas

**Vantagens:**

* **Fácil de aprender e usar:** A sintaxe simples e legível do Python a torna acessível a iniciantes.
* **Extensível:** A biblioteca padrão e o ecossistema de terceiros fornecem uma ampla gama de funcionalidades.
* **Orientado a objetos:** Promove o desenvolvimento modular e flexível.
* **Multiparadigma:** Suporta diferentes paradigmas de programming, como orientado a objetos, funcional e imperativo.
* **Dinamicamente tipado:** Oferece maior conveniência e flexibilidade.

**Desvantagens:**

* **Desempenho:** Pode ser mais lento que linguagens compiladas para grandes conjuntos de dados ou cargas de trabalho intensivas.
* **Gerenciamento de memória:** Sendo uma linguagem interpretada, o Python pode ter problemas de eficiência de memória em comparação com linguagens compiladas.
* **Multiprocessamento limitado:** A concorrência é um desafio em Python, pois o Global Iterator Lock (GIL) pode impedir a execução paralela eficiente.

**Fórmulas Comuns:**

* **Lista de compreensão:** [expressão for item in iterável if condicão]
* **Dicionário de compreensão:** {chave: valor for item in iterável if condicão}
* **Gama:** range(start, stop, step)
* **Módulo de um número:** math.fmod(num1, num2)
* **Fatorial de um número:** math.factorial(num)

Item do edital: Linguagem de programação Scala.

,isDisplayed 食са dentista dentista dentista นี่ \*\*\*\*\*\* ☆ ☆ ☆ ☆ ☆ ☆ ☆ † ☆ derece ♪ ​ ​ ​

Item do edital: Programação funcional.

**Programação Funcional**

A programação funcional é um paradigma de programação que enfatiza:

* **Transformações imutáveis:** As funções não modificam os dados de entrada, mas sim criam novos valores a partir deles.
* **Expressões puras:** O valor de uma expressão é determinado apenas por seus argumentos de entrada, sem efeitos colaterais.
* **Funções de primeira classe:** As funções podem ser passadas como argumentos, retornadas de outras funções e armazenadas em estruturas de dados.

**Características Principais:**

* **Recursão:** Funções podem se chamar recursivamente para resolver problemas menores.
* **Lambdas e funções anônimas:** Pequenas funções sem nome podem ser definidas de forma anônima e passadas como argumentos.
* **Composição:** Funções podem ser compostas para criar novas funções mais complexas.

**Formulações Matemáticas:**

* **Função Pura:** f(x) = y
* **Função Impura:** f(x) = y, onde y pode depender do estado do programa
* **Composição de Funções:** (f ∘ g)(x) = f(g(x))

**Benefícios:**

* **Corretude:** A imutabilidade e a pureza facilitam a garantia de corretude do programa.
* **Paralelismo:** As funções puras podem ser executadas em paralelo sem se preocupar com efeitos colaterais.
* **Modularidade:** As funções de primeira classe permitem uma fácil composição e reutilização de código.

**Aplicações:**

* Processamento de dados
* Aprendizado de máquina
* Desenvolvimento web
* Sistemas distribuídos

**Principais Linguagens:**

* Haskell
* OCaml
* F#
* Scala
* Elixir

Item do edital: Programação orientada a objetos.

**Programação Orientada a Objetos (POO)**

A POO é um paradigma de programação que se concentra em criar objetos que representam entidades do mundo real. Esses objetos contêm dados (atributos) e comportamentos (métodos).

**Princípios Básicos:**

* **Encapsulamento:** Ocultar os detalhes de implementação dentro dos objetos.
* **Abstração:** Expor apenas as informações essenciais e ocultar os detalhes desnecessários.
* **Herança:** Permitir que novas classes (subclasses) herdem atributos e comportamentos de classes existentes (superclasses).
* **Polimorfismo:** Capacidade de objetos de diferentes classes responderem ao mesmo método de maneira diferente.

**Classes e Objetos:**

* **Classe:** Um modelo ou projeto que define os dados e comportamentos de um conjunto de objetos.
* **Objeto:** Uma instância de uma classe que armazena dados e comportamentos específicos.

**Vantagens:**

* **Modularidade:** Facilita a divisão de programas em componentes reutilizáveis.
* **Reusabilidade:** As classes podem ser reutilizadas em diferentes programas.
* **Manutenibilidade:** Alterações em um objeto não afetam outros objetos.
* **Extensibilidade:** Novas funcionalidades podem ser adicionadas facilmente por meio de herança.

**Fórmulas Relacionadas:**

* **Coesão de classe (C):** Medida da quão bem os métodos de uma classe se relacionam com sua responsabilidade primária. C = (Número de métodos altamente acoplados) / (Número de métodos na classe)
* **Acoplamento de classe (D):** Medida de quão fortemente uma classe depende de outras classes. D = (Número de dependências externas) / (Número de dependências totais)

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

**Classes de Objetos**

Em linguagens de programação, os objetos são unidades de dados que possuem dados (atributos) e comportamentos (métodos). As classes são modelos que definem a estrutura e o comportamento desses objetos.

**Tipos de Classes de Objetos em Linguagens de Programação**

* **Vetores:** Coleções lineares de elementos do mesmo tipo. Os elementos são acessados por meio de índices.
* **Listas:** Coleções não ordenadas e mutáveis de elementos de diferentes tipos.
* **Data Frames:** Estruturas de dados tabulares que consistem em linhas e colunas. Cada coluna contém dados homogêneos do mesmo tipo.

**Propriedades**

* **Tipo de Dados:** O tipo de dados dos elementos contidos no objeto.
* **Estrutura:** A forma como os dados são organizados (linear, não ordenado, tabular).
* **Mutabilidade:** Se os dados podem ser modificados após a criação do objeto.
* **Comprimento/Tamanho:** O número de elementos contidos no objeto.
* **Índices:** Os identificadores exclusivos usados para acessar elementos em vetores e listas.
* **Cabeçalhos de Coluna:** Os nomes ou rótulos das colunas em um data frame.
* **Índices de Linha:** Os identificadores exclusivos usados para acessar linhas em um data frame.

**Fórmulas**

* **Comprimento de um vetor ou lista:** len(vetor/lista)
* **Índice de um elemento específico em um vetor ou lista:** vetor/lista[índice]
* **Valor de uma célula em um data frame:** data\_frame[índice\_linha, índice\_coluna]

Item do edital: Manipulação e tabulação de dados com numpy

**Manipulação de Dados com NumPy**

O NumPy fornece funções para criar, manipular e indexar arrays multidimensionais de forma eficiente.

**Criação de Arrays:**

* np.array(dados): Converte uma lista ou tupla em um array NumPy.
* np.zeros(forma): Cria um array preenchido com zeros.
* np.ones(forma): Cria um array preenchido com uns.
* np.full(forma, valor): Cria um array preenchido com um valor específico.

**Manipulação de Arrays:**

* **Operações Aritméticas:** +, -, \*, /, \*\*
* **Operações Relacionais:** ==, !=, <, <=, >, >=
* **Operações Lógicas:** &, |, ~, ^
* **Agregação:** sum(), mean(), std(), max(), min()
* **Transformações:** np.reshape(), np.transpose(), np.flatten()

**Indexação e Slicing:**

* **Indexação Uni e Multidimensional:** array[índice], array[índice1, índice2, ...]
* **Slicing:** array[início:fim:passo], onde início e fim são índices opcionais e passo é o intervalo de indexação.

**Tabulação de Dados com NumPy**

O NumPy pode ser usado para criar tabelas de dados, também conhecidas como matrizes. Ele fornece os seguintes tipos de dados:

* **Arrays Estruturados:** Armazenam dados heterogêneos em campos nomeados.
* **Registros:** Coleções de arrays vinculados por um único índice.
* **Dicionários de Arrays:** Coleções de arrays com chaves de string.

**Fórmulas**

* **Média:** np.mean(array)
* **Desvio Padrão:** np.std(array)
* **Variança:** np.var(array)
* **Covariança:** np.cov(array1, array2)
* **Correlação:** np.corrcoef(array1, array2)

Item do edital: Manipulação e tabulação de dados com pandas

**Manipulação de Dados com Pandas**

O Pandas é uma biblioteca Python projetada para manipulação e análise de dados. Permite operações eficientes em conjuntos de dados tabulares, conhecidos como DataFrames.

**Criação de DataFrames:**

* pd.DataFrame(dados): Cria um DataFrame a partir de uma lista, dicionário ou tupla.
* pd.read\_csv('arquivo.csv'): Lê dados de um arquivo CSV.

**Manipulação de Linhas e Colunas:**

* df.loc[linhas, colunas]: Seleciona linhas e colunas específicas.
* df.iloc[linhas, colunas]: Seleciona linhas e colunas por índice.
* df.drop('coluna'): Remove uma coluna.
* df.rename(colunas): Renomeia colunas.

**Operações de Agregação:**

* df.sum(): Calcula a soma de todos os valores em uma coluna.
* df.mean(): Calcula a média.
* df.max(): Encontra o valor máximo.
* df.min(): Encontra o valor mínimo.

**Filtragem:**

* df[condicao]: Filtra linhas com base em uma condição booleana.
* df.query('condicao'): Filtra linhas com base em uma expressão SQL.

**Operações de Transformação:**

* df.apply(funcao): Aplica uma função a cada elemento do DataFrame.
* df.map(dicionario): Substitui valores com base em um dicionário.
* df.groupby('coluna').mean(): Agrupa linhas por uma coluna e calcula a média.

**Tabulação de Dados:**

O Pandas também fornece recursos para tabulação de dados, permitindo que os usuários resumam e visualizem dados de forma eficaz.

**Tabelas de Frequência:**

* df['coluna'].value\_counts(): Contabiliza ocorrências de valores únicos em uma coluna.

**Tabelas Cruzadas:**

* pd.crosstab('coluna1', 'coluna2'): Cria uma tabela cruzada que conta as ocorrências de valores em duas ou mais colunas.

**Agrupamento e Agregação:**

* df.pivot\_table('coluna\_indice', 'coluna\_valores', aggfunc='soma'): Calcula agregados (por exemplo, soma) para grupos de valores em uma coluna de índice.

Item do edital: Manipulação e tabulação de dados com tidyverse

**Manipulação de Dados com Tidyverse**

O tidyverse é uma coleção de pacotes R projetada para manipular e tabular dados de forma eficiente e concisa. Ele oferece funções que permitem:

* **Selecionar:** Selecionar subconjuntos de dados usando a sintaxe select(var1, var2, ...)
* **Filtrar:** Filtrar dados com base em critérios específicos usando filter(condition1, condition2, ...)
* **Transformar:** Modificar dados usando funções de mutação, como mutate(new\_var = old\_var + 1)
* **Agrupar:** Agrupar dados por variáveis ​​comuns e aplicar operações resumidas (por exemplo, group\_by(var1) %>% summarize(mean = mean(var2)))
* **Unir:** Combinar dados de diferentes fontes usando funções como inner\_join() e left\_join()
* **Separar:** Dividir dados em vários quadros de dados menores usando separate(var1, into = c('new\_var1', 'new\_var2'), sep = ',')

**Tabulação de Dados com Tidyverse**

O tidyverse também oferece funções especializadas para tabular dados:

* **Tabelas resumidas:** Criar tabelas de frequência ou outras tabelas resumidas usando count(), prop.table() ou add\_count(weights)
* **Tabelas cruzadas:** Gerar tabelas cruzadas usando table() ou add\_margins(table, margins = 'both')
* **Tabelas pivô:** Criar tabelas pivô interativas usando pivot\_table(data, values, rows, cols)
* **Gráficos:** Visualizar dados usando pacotes como ggplot2, que oferece uma ampla gama de gráficos personalizáveis

**Fórmulas Básicas**

Algumas fórmulas básicas de manipulação e tabulação de dados com tidyverse incluem:

* select(var1, var2): Seleciona as variáveis var1 e var2 do quadro de dados
* filter(value > 5): Filtra o quadro de dados para valores maiores que 5
* mutate(new\_var = old\_var \* 2): Modifica o quadro de dados criando uma nova variável new\_var como o dobro de old\_var
* group\_by(var1) %>% summarize(mean = mean(var2)): Agrupa o quadro de dados por var1 e calcula a média de var2 para cada grupo
* count(var1, sort = TRUE): Cria uma tabela de frequência mostrando a contagem de valores de var1 em ordem decrescente

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

**Manipulação e Tabelamento de Dados com data.table**

data.table é um pacote R que oferece uma estrutura de dados abrangente e eficiente para manipulação de dados. Ele é uma alternativa ao data.frame padrão, oferecendo recursos adicionais como:

* **Ordenação por colunas múltiplas:** data.table[order(col1, col2)].
* **Joins rápidos:** data.table[j(outra\_tabela)].
* **Agregações flexíveis:** data.table[, sum(valor), by=agrupamento].
* **Seleção de colunas eficiente:** data.table[, .(col1, col2)].
* **Filtragem avançada:** data.table[condicao1 | condicao2].

**Vantagens:**

* **Performance:** A estrutura de dados otimizada resulta em operações mais rápidas.
* **Facilidade de uso:** A sintaxe intuitiva facilita o aprendizado e uso.
* **Extensibilidade:** Pode ser estendido com pacotes adicionais para funcionalidade avançada.

**Exemplos:**

# Cria uma data.table

dt <- data.table(id = c(1, 2, 3), nome = c("João", "Maria", "Pedro"))

# Ordena por nome

dt[order(nome)]

# Junta com outra tabela

dt[j(outra\_tabela, by="id")]

# Agrupa por id e calcula a soma do valor

dt[, sum(valor), by=id]

# Seleciona as colunas id e nome

dt[, .(id, nome)]

# Filtra registros com id maior que 2

dt[id > 2]

**Conclusão:**

data.table é uma ferramenta poderosa para manipulação e tabelamento de dados em R. Ele oferece recursos avançados, performance aprimorada e facilidade de uso, tornando-o uma escolha excelente para tarefas de gerenciamento de dados.

Item do edital: Visualização de dados com ggplot

**Visualização de Dados com ggplot**

O ggplot é uma biblioteca de visualização de dados R que permite a criação de gráficos complexos e esteticamente agradáveis com facilidade. Ele se baseia no sistema gramatical de gráficos (GGS), onde os dados são representados como um conjunto de camadas que podem ser combinadas e modificadas para criar gráficos personalizados.

**Fórmulas de Geometria:**

As fórmulas de geometria são o núcleo do ggplot. Elas definem a geometria dos dados a serem exibidos, como pontos, linhas ou barras.

* geom\_point(): Cria um gráfico de dispersão de pontos.
* geom\_line(): Cria um gráfico de linhas.
* geom\_bar(): Cria um gráfico de barras.
* geom\_histogram(): Cria um histograma.
* geom\_density(): Cria uma curva de densidade estimada.

**Fórmulas de Estatística:**

As fórmulas de estatística definem os cálculos estatísticos a serem aplicados aos dados.

* stat\_summary(): Fornece estatísticas resumidas (por exemplo, média, mediana) como texto ou formas geométricas.
* stat\_smooth(): Ajusta uma linha ou curva suavizada nos dados.
* stat\_bin() Bin agrupa os dados em intervalos e cria um histograma ou curva de densidade.

**Fórmulas de Eixos e Legendas:**

As fórmulas de eixos e legendas controlam a aparência dos eixos e legendas do gráfico.

* labs(title = "", x = "", y = ""): Define os títulos dos eixos e do gráfico.
* scale\_x\_continuous(): Controla o intervalo e as divisões do eixo x.
* scale\_y\_discrete(): Controla o intervalo e as divisões do eixo y.
* legend(): Adiciona uma legenda ao gráfico.

**Exemplo de Código:**

Para criar um gráfico de dispersão mostrando a relação entre altura e peso:

library(ggplot2)

ggplot(data = df,

mapping = aes(x = height, y = weight)) +

geom\_point() +

labs(title = "Relação entre Altura e Peso")

Item do edital: Visualização de dados com matplotlib.

**Visualização de Dados com Matplotlib**

Matplotlib é uma biblioteca Python popular para visualização de dados que permite criar gráficos estáticos, animados e interativos.

**Recursos:**

* **Variedade de tipos de gráfico:** incluindo barras, linhas, dispersão, histograma etc.
* **Customização flexível:** controle sobre cores, estilos de linha, tamanhos de fonte etc.
* **Funcionalidades avançadas:** legendas, títulos, eixos, anotações etc.
* **Integração com outras bibliotecas:** como NumPy, Pandas e SciPy para manipulação e análise de dados.

**Fórmulas Básicas:**

* **Gráfico de Linhas:** plt.plot(x, y)
* **Gráfico de Barras:** plt.bar(x, y)
* **Histograma:** plt.hist(data)

**Etapas de Visualização de Dados:**

1. **Importe Matplotlib:** import matplotlib.pyplot as plt
2. **Crie um objeto de figura:** fig, ax = plt.subplots()
3. **Adicione dados ao gráfico:** Use funções como plt.plot() ou plt.bar()
4. **Configure o gráfico:** Defina rótulos, legendas, limites de eixos etc.
5. **Exiba o gráfico:** plt.show()

**Exemplo de Código:**

import matplotlib.pyplot as plt

x = [1, 2, 3]

y = [2, 4, 6]

plt.plot(x, y)

plt.xlabel("x")

plt.ylabel("y")

plt.title("Gráfico de Linhas")

plt.show()

**Conclusão:**

Matplotlib é uma ferramenta poderosa para visualização de dados que permite criar representações gráficas personalizadas e informativas de dados. Sua versatilidade e facilidade de uso a tornam adequada para vários casos de uso, desde análises exploratórias até relatórios de apresentação.

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

**Paralelização de Rotinas de Ciência de Dados**

A paralelização de rotinas de ciência de dados visa melhorar o desempenho e reduzir o tempo de execução distribuindo tarefas entre vários núcleos ou nós de computação. Ela pode ser alcançada usando técnicas como:

* **Multiprocessamento:** Cria processos separados para tarefas paralelas.
* **Multithreading:** Cria threads dentro de um único processo para tarefas paralelas.
* **Computação em Nuvem:** Alavanca serviços de nuvem para executar tarefas em nós de computação paralelos.

**Benefícios:**

* Tempo de execução mais rápido
* Melhor utilização de recursos computacionais
* Processamento de grandes volumes de dados mais eficiente

**Fórmula para Eficiência de Paralelização:**

A lei de Amdahl quantifica a eficiência da paralelização:

Eficiência = 1 / ((1 - P) + P / N)

Onde:

* P é a porcentagem da tarefa que pode ser paralelizada
* N é o número de núcleos ou nós de computação

Quanto maior o valor de P e menor o valor de N, maior será a eficiência da paralelização.

**Considerações:**

* **Sobrecarga de Comunicação:** A comunicação entre processos ou threads pode introduzir sobrecarga.
* **Tarefas Dependentes:** Tarefas com dependências podem não ser apropriadas para paralelização.
* **Escolhendo a Melhor Técnica:** A técnica de paralelização apropriada depende do tipo de tarefa e da infraestrutura computacional disponível.

**Exemplo:**

Suponha que uma tarefa possa ser 80% paralelizada. Ao dividir a tarefa em 8 núcleos, a eficiência de paralelização seria:

Eficiência = 1 / ((1 - 0,8) + 0,8 / 8) = 0,92

Isso indica que a paralelização melhoraria o tempo de execução em aproximadamente 92%.

Item do edital: Probabilidade e probabilidade condicional.

**Probabilidade**

Probabilidade é uma medida numérica que quantifica a probabilidade de ocorrência de um evento. É expressa como um valor entre 0 e 1, onde:

* **0:** O evento é impossível.
* **1:** O evento é certo.
* **Valores entre 0 e 1:** O evento tem probabilidade variável.

**Probabilidade Condicional**

A probabilidade condicional mede a probabilidade de um evento ocorrer, dado que outro evento já ocorreu. É representada por P(A|B), onde:

* **P(A)** é a probabilidade do evento A.
* **P(B)** é a probabilidade do evento B.
* **P(A|B)** é a probabilidade de A ocorrer, dado que B ocorreu.

**Fórmula para Probabilidade Condicional**

P(A|B) = P(A ⋂ B) / P(B)

onde:

* **P(A ⋂ B)** é a probabilidade da interseção de A e B, ou seja, a probabilidade de ambos os eventos ocorrerem.

**Exemplo**

Suponha que você esteja lançando uma moeda. A probabilidade de cara é 1/2. A probabilidade de coroa também é 1/2.

Se você lançar a moeda e obter coroa, a probabilidade condicional de cara no próximo lançamento é:

P(cara | coroa) = P(cara ⋂ coroa) / P(coroa)

Como a probabilidade de cara e coroa é independente, a interseção é nula. Portanto:

P(cara | coroa) = 0 / 1/2

**Consequências**

A probabilidade condicional é fundamental para:

* Prever eventos com base em eventos anteriores.
* Tomar decisões informadas.
* Desenvolver modelos estatísticos.

Item do edital: Independência de eventos

**Independência de Eventos**

Dois ou mais eventos são considerados independentes se a ocorrência de um evento não afeta a probabilidade de ocorrência de qualquer outro evento.

**Definição Matemática**

Eventos A e B são independentes se:

P(A ∩ B) = P(A) \* P(B)

onde:

* P(A ∩ B) é a probabilidade da interseção de A e B
* P(A) é a probabilidade de A
* P(B) é a probabilidade de B

**Propriedades da Independência**

Se A e B são independentes:

* A e B são também independentes de qualquer subconjunto de A ou B
* Eventos múltiplos (A, B, C, ..., N) são todos independentes entre si
* A probabilidade da união de eventos independentes é dada por:

P(A ∪ B) = P(A) + P(B) - P(A ∩ B)

**Importância**

A independência de eventos é crucial para calcular probabilidades em cenários complexos. Permite decompor problemas em eventos menores e independentes, tornando os cálculos mais fáceis.

**Exemplos**

* Lançar uma moeda duas vezes: Cada lançamento é independente do outro, pois o resultado de um não afeta o do outro.
* Escolher duas cartas de um baralho: O valor da segunda carta escolhida é independente da primeira, pois o baralho é embaralhado entre as escolhas.

Item do edital: teorema de Bayes

**Teorema de Bayes**

O Teorema de Bayes é uma regra matemática que permite atualizar as probabilidades à luz de novas evidências. É amplamente utilizado em estatística, aprendizado de máquina e teoria de decisão.

**Formulação**

Seja X uma variável aleatória com espaço amostral S e seja B um evento no espaço amostral. Então, a probabilidade de B dado X = x é dada por:

P(B | X = x) = (P(X = x | B) \* P(B)) / P(X = x)

onde:

* P(B | X = x) é a probabilidade posterior de B dado X = x
* P(X = x | B) é a probabilidade condicional de X = x dado B
* P(B) é a probabilidade de B
* P(X = x) é a probabilidade marginal de X = x

**Interpretação**

O Teorema de Bayes permite que você atualize sua crença sobre a probabilidade de um evento à luz de novas informações. O termo "posterior" refere-se à probabilidade atualizada, enquanto o termo "prior" refere-se à probabilidade original.

**Aplicações**

O Teorema de Bayes tem uma ampla gama de aplicações, incluindo:

* Classificação e previsão: Usado para determinar a probabilidade de um ponto de dados pertencer a uma determinada classe ou categoria.
* Diagnóstico médico: Usado para calcular a probabilidade de uma pessoa ter uma determinada doença com base em seus sintomas.
* Análise de risco: Usado para avaliar o risco de um evento futuro ocorrendo.

**Fórmulas Relacionadas**

* **Lei das Probabilidades Totais:** P(X = x) = Σ(P(X = x | B) \* P(B)) para todos os B em S
* **Probabilidade Condicional:** P(X = x | B) = P(B | X = x) \* P(X = x) / P(B)

Item do edital: teorema da probabilidade total.

**Teorema da Probabilidade Total**

O Teorema da Probabilidade Total, também conhecido como Lei da Probabilidade Total, determina a probabilidade de um evento ocorrer quando existem vários eventos mutuamente exclusivos que cobrem todo o espaço amostral.

**Fórmula:**

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

onde:

* P(A) é a probabilidade do evento A ocorrer
* B1, B2, ..., Bn são eventos mutuamente exclusivos e exaustivos que cobrem o espaço amostral
* P(A|Bi) é a probabilidade condicional de A ocorrer dado que Bi ocorreu
* P(Bi) é a probabilidade do evento Bi

**Interpretação:**

O teorema afirma que a probabilidade total de um evento A pode ser calculada como a soma das probabilidades de A ocorrer em cada um dos eventos mutuamente exclusivos B1, B2, ..., Bn multiplicadas pelas probabilidades dos eventos B.

**Exemplo:**

Suponha que você tenha um baralho de cartas com 52 cartas e esteja interessado em tirar uma carta vermelha. Os eventos mutuamente exclusivos e exaustivos são:

* B1: Tirar uma carta de copas (13 cartas)
* B2: Tirar uma carta de ouros (13 cartas)

A probabilidade de tirar uma carta vermelha é:

P(Vermelha) = P(Vermelha|Copas)P(Copas) + P(Vermelha|Ouros)P(Ouros) = (1/13)(1/4) + (1/13)(1/4) = 1/26

Portanto, a probabilidade de tirar uma carta vermelha de um baralho padrão de 52 cartas é de 1/26.

Item do edital: Variáveis aleatórias e funções de probabilidade.

**Variáveis Aleatórias**

Uma variável aleatória é uma variável que assume valores aleatórios de acordo com uma determinada distribuição de probabilidade. Ela representa o resultado de um experimento aleatório ou evento incerto.

**Funções de Probabilidade**

Uma função de probabilidade define a probabilidade de uma variável aleatória assumir um valor específico ou intervalo de valores. Para uma variável aleatória discreta, ela é dada por:

P(X = x) = probabilidade da variável aleatória assumir o valor x

Para uma variável aleatória contínua, ela é dada por:

f(x) = probabilidade por unidade de x da variável aleatória assumir o valor x

**Características das Funções de Probabilidade**

As funções de probabilidade devem satisfazer as seguintes propriedades:

* Não negatividade: P(X = x) >= 0 ou f(x) >= 0 para todos os valores de x
* Normalização: A soma das probabilidades de todos os valores possíveis da variável aleatória é 1, ou seja, ΣP(X = x) = 1 ou ∫f(x)dx = 1

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

**Distribuições de Probabilidade Discretas e Contínuas**

**Distribuições de Probabilidade Discretas:**

* Representam variáveis aleatórias que podem assumir valores distintos e separados.
* A probabilidade de cada valor é calculada utilizando a função de massa de probabilidade (PMF).
* Exemplos: distribuição binomial, distribuição de Poisson e distribuição geométrica.

**Distribuições de Probabilidade Contínuas:**

* Representam variáveis aleatórias que podem assumir qualquer valor dentro de um intervalo ou conjunto contínuo.
* A probabilidade de um intervalo específico é calculada utilizando a função de densidade de probabilidade (PDF).
* Exemplos: distribuição normal, distribuição t de Student e distribuição uniforme.

**Distribuição Uniforme:**

**Definição:** Uma distribuição uniforme descreve uma variável aleatória que pode assumir qualquer valor dentro de um intervalo definido [a, b] com probabilidade igual.

**Função de Densidade de Probabilidade (PDF):**

f(x) = 1 / (b - a) para a <= x <= b

**Função de Distribuição Cumulativa (CDF):**

F(x) = (x - a) / (b - a) para a <= x <= b

**Características:**

* Todos os valores dentro do intervalo têm a mesma probabilidade de ocorrência.
* É uma distribuição não simétrica.
* A média e a mediana são (a + b) / 2.
* O desvio padrão é (b - a) / sqrt(12).

**Aplicações:**

* Modelagem de variáveis aleatórias que representam valores selecionados aleatoriamente dentro de um intervalo.
* Simulações, geração de números aleatórios e programação de Monte Carlo.

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

**Distribuições de Probabilidade**

As distribuições de probabilidade descrevem a probabilidade de ocorrência de diferentes valores em uma variável aleatória. Elas podem ser discretas ou contínuas.

**Distribuições Discretas**

* Valores ocorrem em pontos específicos (distintos).
* A probabilidade de cada valor é dada por uma função de massa de probabilidade (PMF).

**Distribuições Contínuas**

* Valores podem assumir qualquer valor dentro de um intervalo.
* A probabilidade de um valor específico é zero, mas a probabilidade de um intervalo de valores é dada por uma função de densidade de probabilidade (PDF).

**Distribuição Binomial**

A distribuição binomial é uma distribuição discreta que descreve o número de sucessos em um número fixo de tentativas independentes com probabilidade constante de sucesso p. Sua PMF é dada por:

P(X = x) = (n! / x!(n-x)!) \* p^x \* (1-p)^(n-x)

onde:

* X é a variável aleatória que representa o número de sucessos
* n é o número de tentativas
* p é a probabilidade de sucesso em cada tentativa

**Características da Distribuição Binomial**

* Média: μ = n \* p
* Variância: σ^2 = n \* p \* (1-p)

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

**Distribuições de Probabilidade Discretas e Contínuas**

**Distribuições Discretas**

* Representam variáveis aleatórias que assumem valores distintos e separados.
* Exemplo: número de clientes que chegam a uma loja em um determinado período de tempo.

**Distribuições Contínuas**

* Representam variáveis aleatórias que podem assumir qualquer valor dentro de um determinado intervalo.
* Exemplo: altura de pessoas em uma população.

**Distribuição Poisson**

Uma distribuição de probabilidade discreta que modela o número de ocorrências de um evento em um intervalo de tempo ou espaço fixo.

**Fórmula:**

P(X = k) = (e^(-λ) \* λ^k) / k!

onde:

* X é a variável aleatória
* k é o número de ocorrências
* λ é a taxa média de ocorrências

**Características:**

* Média: λ
* Variância: λ
* Valores discretos, não negativos

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

**Distribuições de Probabilidade Discretas**

As distribuições de probabilidade discretas descrevem experimentos com um número finito ou contável de resultados possíveis. Cada resultado tem uma probabilidade específica associada a ele.

* Exemplo: Número de sucessos em um determinado número de lançamentos de moeda

**Distribuições de Probabilidade Contínuas**

As distribuições de probabilidade contínuas descrevem experimentos com um número infinito de resultados possíveis. A probabilidade de qualquer intervalo específico de valores é dada pela área sob a curva de densidade de probabilidade.

* Exemplo: Altura de pessoas selecionadas aleatoriamente

**Distribuição Normal**

A distribuição normal, também conhecida como distribuição gaussiana, é uma distribuição contínua que modela muitos fenômenos naturais. Sua função de densidade de probabilidade é:

f(x) = (1 / (σ√(2π))) \* e^(-(x - μ)² / (2σ²))

onde:

* x é o valor da variável aleatória
* μ é a média
* σ é o desvio padrão

**Características da Distribuição Normal:**

* É simétrica em torno da média
* O desvio padrão determina a largura da distribuição
* 68% dos dados estão dentro de um desvio padrão da média
* 95% dos dados estão dentro de dois desvios padrão da média
* 99,7% dos dados estão dentro de três desvios padrão da média

Item do edital: Medidas de tendência central

**Medidas de Tendência Central**

Medidas de tendência central resumem e representam um conjunto de dados fornecendo um valor único que descreve seu centro. As principais medidas de tendência central são:

* **Média (X̄)**: A soma de todos os valores dividida pelo número de valores presentes no conjunto de dados.

X̄ = ΣX / n

* **Mediana (Md)**: O valor do meio do conjunto de dados quando classificado em ordem crescente ou decrescente.
* **Moda (Mo)**: O valor que ocorre com mais frequência no conjunto de dados.
* **Média ponderada (X̄w)**: A média calculada utilizando pesos que refletem a importância de cada valor.

X̄w = (Σ(Wi \* Xi)) / ΣWi

* **Média aparada**: A média calculada após remover uma porcentagem fixa de valores mais altos e mais baixos.

**Características das Medidas de Tendência Central**

* **Média:** Sensível a outliers (valores extremos), pois é influenciada por todos os valores.
* **Mediana:** Robusta a outliers, pois não é afetada por valores extremos.
* **Moda:** Pode não ser única ou pode não existir para conjuntos de dados bimodais ou multimodais.
* **Média ponderada:** Considera a importância relativa dos valores.
* **Média aparada:** Reduz o impacto de outliers.

**Quando usar cada medida**

A escolha da medida de tendência central apropriada depende do tipo de dados e do objetivo da análise:

* **Média:** Adequada para dados simétricos e sem outliers.
* **Mediana:** Preferida para dados assimétricos ou com outliers.
* **Moda:** Útil para identificar o valor mais comum ou identificar padrões em dados categóricos.
* **Média ponderada:** Quando os valores têm pesos diferentes.
* **Média aparada:** Quando os outliers precisam ser mitigados.

Item do edital: Medidas de dispersão

**Medidas de Dispersão**

As medidas de dispersão quantificam a variabilidade ou espalhamento dos dados em torno de uma medida central (como média ou mediana). Elas medem o quão espalhados os dados estão em relação a um valor central, fornecendo uma indicação da heterogeneidade ou homogeneidade do conjunto de dados.

**Tipos de Medidas de Dispersão**

* **Amplitude (ou Escopo):** Diferença entre os valores máximo e mínimo.
  + Fórmula: Amplitude = Valor máximo - Valor mínimo
* **Desvio Médio Absoluto (DMA):** Média das diferenças absolutas entre cada valor de dados e a mediana.
  + Fórmula: DMA = (1/n) \* ∑|X - M|, onde X é o valor dos dados e M é a mediana
* **Variância:** Média das diferenças quadradas entre cada valor de dados e a média. Mede o quão espalhados os dados estão em relação à média.
  + Fórmula: Variância = (1/n) \* ∑(X - μ)²
* **Desvio Padrão:** Raiz quadrada da variância. É proporcional à amplitude dos dados e mede a dispersão em termos de unidades originais.
  + Fórmula: Desvio Padrão = √(Variância)
* **Coeficiente de Variação (CV):** Variância dividida pela média. Mede a dispersão relativa dos dados.
  + Fórmula: CV = (Desvio Padrão / Média) \* 100%

**Interpretação**

Medidas de dispersão mais altas indicam que os dados são mais espalhados, enquanto medidas mais baixas indicam que os dados são mais concentrados em torno da medida central.

A escolha da medida de dispersão apropriada depende da natureza dos dados, dos objetivos da análise e das distribuições estatísticas subjacentes.

Item do edital: Medidas de correlação.

**Medidas de Correlação**

As medidas de correlação quantificam a força e a direção da relação linear entre duas variáveis. Elas medem o grau em que as variáveis aumentam ou diminuem juntas. Existem vários tipos de medidas de correlação:

**Coeficiente de Correlação de Pearson (r)**

Esta é a medida de correlação mais comumente usada. Mede a correlação linear entre duas variáveis contínuas e varia entre -1 e 1:

* **-1:** Correlação perfeita negativa (as variáveis aumentam e diminuem juntas perfeitamente)
* **0:** Sem correlação (nenhuma relação linear)
* **1:** Correlação perfeita positiva (as variáveis aumentam e diminuem juntas perfeitamente)

**Fórmula:**

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

onde:

* x e y são os valores das duas variáveis
* x̄ e ȳ são as médias das duas variáveis

**Coeficiente de Correlação de Spearman (ρ)**

Esta medida é usada para correlacionar duas variáveis ordinais ou contínuas que não são normalmente distribuídas. Ela varia entre -1 e 1, semelhante ao coeficiente de correlação de Pearson.

**Fórmula:**

ρ = 1 - (6∑d²) / (n(n² - 1))

onde:

* d é a diferença nas classificações das duas variáveis
* n é o número de pares de dados

**Coeficiente de Correlação de Kendall (τ)**

Outra medida não paramétrica, o coeficiente de correlação de Kendall também é usado para correlacionar variáveis ordinais. Ele varia entre -1 e 1, indicando uma correlação perfeita negativa ou positiva, respectivamente.

**Fórmula:**

τ = (C - D) / (C + D)

onde:

* C é o número de pares concordantes (onde as classificações de ambas as variáveis concordam)
* D é o número de pares discordantes (onde as classificações das duas variáveis discordam)

Item do edital: Teorema do limite central.

hâ hâ€™›

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

**Regra Empírica (Regra de Três Sigma)**

A regra empírica afirma que, para uma distribuição normal:

* Cerca de 68% dos dados caem **dentro de 1 desvio padrão** da média (µ ± 1σ).
* Cerca de 95% dos dados caem **dentro de 2 desvios padrão** da média (µ ± 2σ).
* Cerca de 99,7% dos dados caem **dentro de 3 desvios padrão** da média (µ ± 3σ).

**Fórmulas:**

* **Desvio padrão (σ):** Medida da dispersão dos dados em torno da média.
* **Média (µ):** Valor médio dos dados.
* **Pontuação z:** Medida de quantas unidades de desvio padrão um valor está distante da média:

Z = (x - µ) / σ

**Onde:**

* x é o valor do dado
* µ é a média
* σ é o desvio padrão

**Como usar a regra empírica:**

1. Calcule a média e o desvio padrão da distribuição.
2. Para determinar a porcentagem de dados dentro de um determinado número de desvios padrão, use os intervalos da regra empírica:
   * Dentro de 1σ: 68%
   * Dentro de 2σ: 95%
   * Dentro de 3σ: 99,7%
3. Transforme valores individuais em pontuações z para determinar sua posição em relação à média.

**Aplicações:**

A regra empírica é amplamente utilizada em estatística para:

* Estimar a probabilidade de um evento ocorrer.
* Fazer inferências sobre a população a partir de uma amostra.
* Testar hipóteses sobre os parâmetros da distribuição.

Item do edital: Diagramas causais: grafos acíclicos dirigidos

**Diagramas Causais:**

* Gráficos acíclicos dirigidos (DAGs) que representam relações de causa e efeito.
* Os nós representam variáveis, enquanto as setas representam relações causais entre as variáveis.
* Um DAG é acíclico, ou seja, não contém ciclos (caminhos que retornam ao nó inicial).

**Características:**

* **Direcionalidade:** As setas indicam a direção da relação causal, de causa para efeito.
* **Aciclicidade:** Os gráficos não contêm ciclos, evitando paradoxos causais.
* **Hierarquia:** Os nós podem ser organizados hierarquicamente, com variáveis de nível superior influenciando variáveis de nível inferior.

**Usos:**

* Identificar causas potenciais de um resultado.
* Testar hipóteses sobre relações causais.
* Prever resultados com base em relações causais conhecidas.

**Fórmulas:**

* **Fórmula de Probabilidade Condicional:** P(E | C) = P(C | E) \* P(E) / P(C)
* **Fórmula da Cadeia de Markov:** P(X\_n | X\_1, ..., X\_n-1) = P(X\_n | X\_n-1)

Item do edital: Diagramas causais: variáveis confundidoras

**Diagramas Causais: Variáveis Confundidoras**

Variáveis confundidoras são variáveis que influenciam tanto a variável de exposição quanto a variável de desfecho, potencialmente distorcendo a associação observada entre exposição e desfecho.

**Princípios:**

* Uma variável confundidora não deve ser afetada pela exposição.
* Uma variável confundidora deve estar associada à variável de desfecho.
* Uma variável confundidora deve ser associada à variável de exposição.

**Fórmulas:**

**Fórmula de exposição bruta:**

Risco ou Razão de Probabilidade (RP ou RR) = (Proporção de doentes expostos) / (Proporção de não doentes expostos)

**Fórmula ajustada:**

RP ou RR ajustada = RP bruta / (Fator de confusão)

**Onde:**

* Fator de confusão = Razão de Chances (RC) da variável confundidora para o desfecho entre indivíduos expostos / Razão de Chances da variável confundidora para o desfecho entre indivíduos não expostos

**Tipos de Variáveis Confundidoras:**

* **Confundidoras de medida:** Variáveis que são medidas com erro podem atuar como confundidoras se o erro for diferencial em relação à exposição.
* **Confundidoras não medidas:** Variáveis que não são medidas no estudo, mas estão associadas tanto à exposição quanto ao desfecho.
* **Confundidoras de seleção:** Variáveis que influenciam a seleção de indivíduos para o estudo e estão associadas tanto à exposição quanto ao desfecho.

**Ajustamento para Variáveis Confundidoras:**

* Estratficação ou randomização para a variável confundidora.
* Regressão múltipla para controlar o efeito da variável confundidora.
* Restrição de análises a subgrupos onde a variável confundidora é uniforme.

Item do edital: Diagramas causais: variáveis colisoras

**Diagramas Causais: Variáveis Colisoras**

Os diagramas causais são representações gráficas que mostram relações causais entre variáveis. Uma **variável colisor** é uma variável não observada que influencia tanto a variável de exposição quanto a variável de desfecho, distorcendo a relação aparente entre elas.

**Efeitos**

* Variáveis colisoras podem criar **confusão** estatística, fazendo com que pareça que existe uma relação causal quando não existe.
* Elas podem também **mascarar** relações causais reais, fazendo com que pareçam não existir.

**Identificação**

Variáveis colisoras podem ser identificadas por:

* **Critério de back-door:** A variável colisor está relacionada causalmente tanto à variável de exposição quanto à variável de desfecho e não há nenhum caminho causal da variável de exposição à variável de desfecho que não passe pela variável colisor.

**Fórmulas**

A fórmula para o efeito colisor é:

P(Y | X) = ∑\_Z P(Y | X, Z) P(Z | X)

onde:

* Y é a variável de desfecho
* X é a variável de exposição
* Z é a variável colisor

**Controle**

Para controlar as variáveis colisoras, é necessário:

* **Medir** a variável colisor
* **Ajustar** para a variável colisor na análise estatística (por exemplo, usando regressão múltipla)
* **Bloquear** a variável colisor por estratificação ou emparelhamento

Item do edital: Diagramas causais: variáveis de mediação.

**Diagrama de Mediação**

Um diagrama de mediação representa relações entre três ou mais variáveis, onde uma variável (mediadora) transmite o efeito de uma variável (independente) sobre outra (dependente).

**Variáveis de Mediação**

* **Variável Independente:** A variável que inicia o processo.
* **Variável Mediadora:** A variável que intervém entre a variável independe e dependente, transmitindo os efeitos.
* **Variável Dependente:** A variável que é afetada pela variável mediadora e pela variável independe.

\**Fórmulas*

**Efeito Total:** Representa o efeito direto da variável independe sobre a variável dependente.

**Efeito Direto:** Representa o efeito residual da variável independe sobre a variável dependente, após controlar o efeito da variável mediadora.

**Efeito Indireto:** Representa o efeito da variável independe sobre a variável dependente, mediado pela variável mediadora.

**Efeito Moderado da Mediação:** O efeito da variável mediadora varia de acordo com outra variável (variável moderadora).

**Exemplos de Variáveis de Mediação**

* Autoeficácia em relação ao desempenho acadêmico
* Estresse relacionado ao trabalho em relação à satisfação no trabalho
* Motivação intrínseca em relação à persistência no trabalho

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

**Métodos Experimentais RCT (Randomized Controlled Trials)**

**Introdução:** Os Ensaios Clínicos Randomizados (RCTs) são o método gold standard para identificar causalidade em pesquisas médicas e outras áreas. Eles envolvem a randomização de participantes para receber uma intervenção ou um controle, permitindo comparações válidas entre os grupos.

**Conceito Básico:** Em um RCT, os participantes são aleatoriamente designados para receber a intervenção ou o controle. Isso elimina fatores de confusão e garante que os grupos sejam semelhantes em todas as características observadas e não observadas.

**Vantagens:**

* **Alta validade interna:** A randomização elimina o viés de seleção, garantindo que os grupos sejam comparáveis no início.
* **Controle preciso:** O experimentador pode controlar estritamente a intervenção e o controle, reduzindo a variação e aumentando a precisão.
* **Generalização:** Os resultados podem ser generalizados para a população à qual os participantes pertencem.

**Técnicas:**

* **Alocação:** Os participantes são aleatoriamente alocados para os grupos de intervenção e controle.
* **Cegamento:** Os participantes e pesquisadores podem ser cegados em relação à alocação do tratamento para evitar viés.
* **Análise Estatística:** Os dados são analisados usando testes estatísticos apropriados para comparar os desfechos entre os grupos.

**Fórmulas:**

* **Teste-t:** Para comparar médias entre dois grupos independentes.
* **Teste de qui-quadrado:** Para comparar proporções entre grupos.
* **Análise de Variância (ANOVA):** Para comparar médias entre vários grupos.

**Limitações:**

* **Custo e tempo:** Os RCTs podem ser caros e demorados de realizar.
* **Viés potencial:** Embora a randomização reduza o viés, outros tipos de viés podem ocorrer, como viés de adesão ou perda de participantes.
* **Generalização limitada:** Os resultados podem não ser generalizáveis para todas as populações devido a diferenças nas características dos participantes.

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

**Métodos de Identificação Quase-Experimental**

Os métodos quase-experimentais são utilizados para estabelecer relações causais quando a randomização não é viável. Eles envolvem controlar variáveis externas e comparar grupos quase-equivalentes para estimar os efeitos da intervenção.

**Tipos de Métodos Quase-Experimentais**

* **Grupos de Comparação Não Equivalentes:** Comparação de grupos pré-existentes que diferem em características observáveis. Métodos:
  + **Desenho de Série Temporal Interrompida:** Medição de resultados antes e depois de uma intervenção, controlando por tendências temporais.
* **Desenhos de Regressão Descontínua:** Atribuição do tratamento com base em uma variável de corte, como renda ou idade. Métodos:
  + **Regressão Descontínua Regressiva:** Variável de corte contínua.
  + **Regressão Descontínua Sharp:** Variável de corte dicotômica.
* **Designs de Variáveis Instrumentais:** Uso de uma variável instrumental, que influencia a atribuição do tratamento, mas não os resultados diretamente. Método:
  + **Variável Instrumental de Dois Estágios:**
    1. Regressão da atribuição de tratamento na variável instrumental.
    2. Regressão dos resultados na atribuição de tratamento ajustada.
* **Designs de Combinação:** Combinação de métodos quase-experimentais para melhorar a precisão. Exemplos:
  + **Desenho de Diferenças em Diferenças:** Comparação de mudanças nos resultados entre grupos de tratamento e controle ao longo do tempo.
  + **Desenho de Ponto de Regressão Descontínuo com Diferenças em Diferenças:** Combina regressão descontínua com diferenças em diferenças.

**Fórmulas**

* **Diferença em Diferenças:**

(Y1\_p - Y0\_p) - (Y1\_c - Y0\_c)

onde: \* Y1\_p: Resultado pós-tratamento no grupo de tratamento \* Y0\_p: Resultado pré-tratamento no grupo de tratamento \* Y1\_c: Resultado pós-tratamento no grupo de controle \* Y0\_c: Resultado pré-tratamento no grupo de controle

* **Regressão Descontínua Sharp:**

Y = α + β1(T) + β2(Z) + ε

onde: \* Y: Resultado \* T: Tratamento (0 = controle, 1 = tratamento) \* Z: Variável de corte \* α, β1, β2, ε: Parâmetros do modelo

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

**Viés de Amostragem**

**Definição:**

Viés de amostragem ocorre quando uma amostra não representa adequadamente a população da qual foi extraída. Isso pode levar a conclusões erradas sobre a população.

**Tipos de Viés de Amostragem:**

* **Amostragem de conveniência:** Selecionar participantes que são fáceis de acessar, o que pode levar a uma representação tendenciosa.
* **Amostragem voluntária:** Recrutar participantes que se voluntariam, o que pode resultar em uma amostra com características específicas que não são representativas da população.
* **Amostragem intencional:** Selecionar participantes com base em critérios específicos, o que pode levar a uma amostra tendenciosa se os critérios não representarem a população.
* **Amostragem em bola de neve:** Recrutar participantes perguntando aos participantes atuais, o que pode levar a uma amostra concentrada em grupos específicos.

**Fórmulas:**

O viés de amostragem pode ser calculado como a diferença entre a proporção de uma característica na população e a proporção na amostra:

Viés = (Proporção na população) - (Proporção na amostra)

**Soluções:**

* Escolher um método de amostragem que garanta que a amostra representa a população.
* Usar técnicas de ponderação para ajustar a amostra para torná-la mais representativa.
* Coletar dados de várias fontes para reduzir o impacto de um único viés.
* Reconhecer as limitações da amostra e considerar seu impacto nas conclusões.

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

**Viés de Seleção**

**Definição:** O viés de seleção ocorre quando uma amostra não representa adequadamente a população-alvo devido ao processo de seleção.

**Causas:**

* Recrutamento seletivo: Participação desigual de diferentes grupos na coleta de dados.
* Amostragem tendenciosa: Seleção de participantes com base em critérios que afetam a variável de interesse.
* Perdas durante o acompanhamento: Participação diferencial na coleta de dados entre grupos diferentes.

**Efeitos:**

* Estimativas tendenciosas da prevalência ou distribuição da variável de interesse.
* Conclusões imprecisas sobre as relações entre as variáveis.

**Fórmulas:**

* **Probabilidade de seleção:** P(S) = (Número de indivíduos selecionados) / (Tamanho total da população)
* **Probabilidade de inclusão:** P(I) = P(S) \* P(Concluiu a fase 1) \* ... \* P(Concluiu a fase n)

**Soluções:**

* **Amostragem aleatória:** Sorteio aleatório de participantes para garantir a representação de todos os grupos.
* **Amostragem estratificada:** Divisão da população em subgrupos e amostragem aleatória dentro de cada subgrupo.
* **Pesos de amostragem:** Ajuste dos pesos dos dados para compensar os desvios de seleção.
* **Análise de viés de seleção:** Identificação e quantificação do viés de seleção usando técnicas estatísticas, como modelagem de propensão à pontuação.
* **Métodos de amostragem não probabilística:** Reconhecimento das limitações da amostra e ajuste das análises de acordo.

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

**Viés de desgaste (Attrition bias)**

**Conceito:** O viés de desgaste ocorre quando os sujeitos de um estudo abandonam ou são removidos da amostra, e esses sujeitos são sistematicamente diferentes dos que permanecem. Isso pode levar a resultados distorcidos, pois as características e comportamentos dos sujeitos que desistem podem influenciar os resultados.

**Soluções:**

* **Análise de razões de probabilidade:** Modele a probabilidade de um sujeito desistir com base em suas características observadas para ajustar os resultados.
* **Imputação de valores ausentes:** Substitua os dados ausentes de indivíduos que desistiram por valores estimados com base nos dados observados dos indivíduos que permaneceram.
* **Estratificação:** Divida a amostra em subgrupos com base no risco de desgaste e analise cada subgrupo separadamente.
* **Reamostragem ponderada:** Atribua pesos aos indivíduos restantes com base na probabilidade de desgaste para compensar os indivíduos que desistiram.
* **Análise de sensibilidade:** Execute análises adicionais sob diferentes suposições sobre o viés de desgaste para avaliar o impacto potencial nos resultados.

**Fórmulas:**

* **Razão de probabilidade:**

log(odds(desistência)) = β0 + β1X1 + ... + βnXn

Onde:

* odds(desistência) é a probabilidade de desistência dividida pela probabilidade de permanência
* X1, ..., Xn são características observadas que podem influenciar a desistência
* β0, ..., βn são coeficientes estimados
* **Peso de reamostragem ponderada:**

w = 1 / P(desistência)

Onde:

* w é o peso do indivíduo
* P(desistência) é a probabilidade estimada de desistência do indivíduo

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

**Viés de Relatório**

**Definição:**

Viés de relatório é um tipo de viés nos dados que ocorre quando os participantes de um estudo relatam informações de forma imprecisa ou incompleta devido a fatores como memória, percepção subjetiva ou motivação.

**Tipos:**

* **Viés de recordação:** Os participantes podem ter dificuldade em lembrar eventos com precisão, especialmente se tiverem ocorrido há muito tempo.
* **Viés de atribuição:** Os participantes podem atribuir incorretamente eventos ou comportamentos a causas específicas.
* **Viés de confirmação:** Os participantes podem relatar informações que confirmam suas crenças ou hipóteses prévias.
* **Viés de resposta:** Os participantes podem alterar suas respostas para agradar ao pesquisador ou evitar avaliações negativas.

**Causas:**

* Memória imprecisa
* Percepções subjetivas
* Motivações ou incentivos
* Fatores contextuais (por exemplo, pressão social)

**Consequências:**

* Dados tendenciosos que podem levar a conclusões erradas
* Dificuldade em generalizar os resultados para a população mais ampla
* Comprometimento da validade interna e externa da pesquisa

**Estratégias de Mitigação:**

* Usar medidas objetivas sempre que possível
* Treinar os entrevistadores para minimizar o viés
* Garantir o anonimato e a confidencialidade
* Triangular dados de várias fontes
* Realizar verificações de consistência interna

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

**Viés de Medição**

**Definição:** Viés de medição ocorre quando um instrumento de medição ou método de coleta de dados introduz erros sistemáticos nos dados. Isso leva a estimativas tendenciosas dos parâmetros do modelo.

**Causas:**

* Erros de instrumento (por exemplo, calibração inadequada)
* Definições imprecisas de variáveis
* Efeito Hawthorne (alteração do comportamento devido à consciência de ser observado)

**Soluções:**

* **Calibração regular:** Verifique e ajuste os instrumentos de medição regularmente para garantir precisão.
* **Padronização:** Use métodos de coleta de dados consistentes para minimizar a variabilidade.
* **Treinamento de avaliadores:** Treine os avaliadores para que usem definições de variáveis precisas e evitem subjetividade.
* **Uso de métricas múltiplas:** Use diferentes instrumentos ou métodos de coleta de dados para confirmar as medições.
* **Aplicação de correções:** Se possível, identifique e corrija os erros de medição usando técnicas estatísticas.

**Fórmulas:**

Não há fórmulas específicas para quantificar o viés de medição. No entanto, as seguintes métricas podem ajudar a detectar e corrigir erros:

* **Coeficiente de variação:** Mede a variabilidade das medições em relação ao seu valor médio.
* **Proporção de acordo:** Compara as medições de dois instrumentos ou avaliadores diferentes.
* **Erro absoluto:** Diferença entre uma medição e o valor real (se conhecido).

Item do edital: Modelos probabilísticos gráficos: cadeias de Markov

**Modelos Probabilísticos Gráficos: Cadeias de Markov**

As Cadeias de Markov (MC) são modelos probabilísticos gráficos que representam sistemas nos quais o estado futuro depende apenas do estado presente. Elas são frequentemente usadas para modelar sequências de eventos ou dados onde o passado contém informações relevantes para prever o futuro.

**Definição Formal:**

Uma Cadeia de Markov de ordem **n** é um processo estocástico com as seguintes propriedades:

* **Estado atual xt:** Representa o estado do sistema no tempo **t**.
* **Conjunto de estados E:** O conjunto de todos os possíveis estados do sistema.
* **Matriz de transição P:** Uma matriz quadrada que especifica a probabilidade de transição de um estado para outro.
* **Probabilidade inicial π:** A probabilidade de estar em um determinado estado no tempo 0.

**Probabilidade de Transição:**

A probabilidade de transição de um estado **i** para um estado **j** em uma Cadeia de Markov de ordem **n** é dada por:

P(x<sub>t+1</sub> = j | x<sub>t</sub> = i) = P<sub>ij</sub>

onde **Pij** é o elemento **(i, j)** da matriz de transição **P**.

**Probabilidade de Estado:**

A probabilidade de estar em um estado **i** no tempo **t** é dada por:

P(x<sub>t</sub> = i) = π<sub>i</sub> \* P(x<sub>t-1</sub> = i<sub>t-1</sub>) \* ... \* P(x<sub>t-n</sub> = i<sub>t-n</sub>)

onde **πi** é a probabilidade inicial de estar em **i** e **it-n** é o estado **n** passos no passado.

**Aplicações:**

As Cadeias de Markov são amplamente utilizadas em vários campos, incluindo:

* Modelagem de linguagem
* Reconhecimento de padrões
* Análise financeira
* Cadeias de fornecimento
* Simulação de sistemas complexos

Item do edital: Modelos probabilísticos gráficos: filtros de Kalman

**Modelos Probabilísticos Gráficos: Filtros de Kalman**

Os filtros de Kalman são um tipo de modelo probabilístico gráfico que estima o estado de um sistema dinâmico linear observando seus comportamentos ao longo do tempo. Eles são usados em uma ampla gama de aplicações, incluindo navegação, previsão do tempo e processamento de sinais.

**Modelo**

Um filtro de Kalman modela um sistema dinâmico como segue:

* **Estado:** O estado do sistema é representado por um vetor **x**.
* **Equação do estado:** O estado evolve com o tempo de acordo com **x[k] = A*x[k-1] + B*u[k] + w[k]**, onde A é a matriz de transição do estado, B é a matriz de controle, u é a entrada e w é o ruído do processo.
* **Equação de observação:** O estado é observado por meio de sensores, resultando nas observações **y[k] = C\*x[k] + v[k]**, onde C é a matriz de observação e v é o ruído de observação.

**Estimativa**

O objetivo do filtro de Kalman é estimar o estado **x[k]** dado as observações **y[1:k]**. Isso é feito usando as seguintes etapas:

* **Predição:** Para prever o estado no próximo passo do tempo, usamos **x[k|k-1] = A*x[k-1|k-1] + B*u[k-1]**.
* **Atualização:** Para atualizar a estimativa após observar **y[k]**, usamos a fórmula de Bayes: **x[k|k] = x[k|k-1] + K\*(y[k] - C\*x[k|k-1])**, onde K é o ganho de Kalman.

**Fórmulas**

* **Ganho de Kalman:** \**K = P[k|k-1]C^T(C*P[k|k-1]*C^T + R)^-1*\*, onde P é a matriz de covariância de estimação do estado.
* **Matriz de covariância de previsão:** \**P[k|k-1] = A*P[k-1|k-1]*A^T + Q*\*, onde Q é a matriz de covariância do ruído do processo.
* **Matriz de covariância de atualização:** \**P[k|k] = (I - K*C)\*P[k|k-1]\*\*, onde I é a matriz identidade.

**Vantagens**

* Eficiente computacionalmente.
* Maneja efetivamente incertezas tanto no estado quanto nas observações.
* Pode ser generalizado para sistemas não lineares usando o filtro de Kalman estendido (EKF).

Item do edital: Modelos probabilísticos gráficos: Redes bayesianas.

-‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎‎

Item do edital: Testes de hipóteses: teste-z

**Testes de Hipóteses: Teste-Z**

O teste-Z é um teste estatístico usado para determinar se a média de uma população é diferente de um valor especificado. É usado quando a distribuição da população é normal (distribuição gaussiana) e o desvio padrão da população é conhecido.

**Hipóteses:**

* **Hipótese nula (H0):** A média da população é igual ao valor especificado (μ0).
* **Hipótese alternativa (Ha):** A média da população é diferente do valor especificado (μ ≠ μ0).

**Estatística de teste:**

Z = (x̄ - μ0) / (σ / √n)

onde:

* x̄ é a média da amostra
* μ0 é o valor especificado
* σ é o desvio padrão da população
* n é o tamanho da amostra

**Distribuição da estatística de teste:**

Sob H0, Z segue uma distribuição normal padrão (Z ~ N(0, 1)).

**Valor crítico:**

O valor crítico, zα/2, é obtido da distribuição normal padrão, onde α é o nível de significância. Os valores críticos dividem a distribuição em duas regiões: a região de rejeição (rejeitar H0) e a região de não rejeição (não rejeitar H0).

**Tomada de decisão:**

* Se |Z| > zα/2, rejeita-se H0.
* Se |Z| ≤ zα/2, não se rejeita H0.

**Interpretação:**

* Se H0 for rejeitada, é evidência estatística de que a média da população é diferente do valor especificado.
* Se H0 não for rejeitada, não há evidência estatística suficiente para concluir que a média da população é diferente do valor especificado.

Item do edital: Testes de hipóteses: teste-t

**Testes de Hipóteses: Teste-t**

Um teste-t é um teste de hipóteses usado para comparar as médias de duas amostras independentes ou pareadas.

**Hipóteses:**

* **Hipótese nula (H0):** As médias das duas amostras são iguais (μ1 = μ2).
* **Hipótese alternativa (Ha):** As médias das duas amostras são diferentes (μ1 ≠ μ2).

**Fórmulas:**

* **Estatística t (amostras independentes):**

t = (x̄1 - x̄2) / √(s²1/n1 + s²2/n2)

onde:

* x̄1 e x̄2 são as médias amostrais
* s²1 e s²2 são as variâncias amostrais
* n1 e n2 são os tamanhos amostrais
* **Estatística t (amostras pareadas):**

t = (d̄ - μ0) / s̄d / √n

onde:

* d̄ é a média das diferenças entre as amostras pareadas
* μ0 é o valor esperado da diferença (geralmente 0)
* s̄d é o desvio padrão das diferenças
* n é o tamanho da amostra

**Procedimento:**

1. **Estabeleça as hipóteses:** Defina a hipótese nula e a hipótese alternativa.
2. **Calcule a estatística t:** Use a fórmula apropriada para amostras independentes ou pareadas.
3. **Determine o valor p:** Use uma tabela de distribuição t ou software estatístico para encontrar o valor p associado à estatística t.
4. **Tome uma decisão:**
   * Se o valor p for menor que o nível de significância (α), rejeite a hipótese nula.
   * Se o valor p for maior ou igual a α, falhe em rejeitar a hipótese nula.

**Observações:**

* Os testes-t assumem que as amostras são normalmente distribuídas.
* Para amostras pequenas (n < 30), é necessário usar uma distribuição t de Student modificada.
* Os testes-t podem ser usados para testar diferenças em proporções ou variâncias, com ajustes nas fórmulas e suposições.

Item do edital: Testes de hipóteses: valor-p

**Testes de Hipóteses: Valor-p**

Um teste de hipóteses é um procedimento estatístico que avalia a evidência contra uma hipótese nula (H0). O valor-p é uma medida que indica quão improvável é obter os resultados observados, assumindo que a hipótese nula é verdadeira.

**Cálculo do Valor-p**

O valor-p é calculado com base na distribuição de probabilidade do teste de hipóteses. Para uma hipótese nula específica, o valor-p é a probabilidade de uma estatística de teste (como a média da amostra) ser tão extrema ou mais extrema do que o valor observado.

**Interpretação do Valor-p**

O valor-p é comparado com um nível de significância (α), que é tipicamente definido como 0,05. Se o valor-p for menor que α, isso significa que existem evidências estatísticas significativas contra a hipótese nula e ela é rejeitada. Se o valor-p for maior que α, não há evidências suficientes para rejeitar a hipótese nula.

**Fórmula para o Valor-p**

Para um teste de hipóteses de média de amostra:

Valor-p = P(X ≥ x | H0)

onde:

* X é a estatística de teste (média da amostra)
* x é o valor observado da estatística de teste
* H0 é a hipótese nula

**Limitações do Valor-p**

O valor-p é uma medida útil, mas tem algumas limitações:

* Ele não fornece a probabilidade de H0 ser verdadeira.
* Ele depende do tamanho da amostra e do efeito do tamanho.
* Pode ser influenciado por fatores não relacionados à hipótese nula.

É importante interpretar o valor-p no contexto do estudo específico e considerar outros fatores ao tomar decisões sobre hipóteses.

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

**Testes de Hipóteses para uma Amostra**

**Introdução:** Testes de hipóteses são procedimentos estatísticos usados para determinar se uma afirmação sobre uma população (hipótese nula) é apoiada por evidências de uma amostra.

**Etapas:**

1. **Definir Hipóteses:**
   * Hipótese Nula (H0): A afirmação a ser testada (geralmente uma igualdade ou desigualdade).
   * Hipótese Alternativa (Ha): A afirmação oposta à H0.
2. **Estabelecer Nível de Significância (α):**
   * Probabilidade de rejeitar H0 quando ela é verdadeira. Tipicamente, α = 0,05.
3. **Selecionar Estatística de Teste:**
   * Estatística que mede a discrepância entre a amostra e a H0. A escolha da estatística depende do tipo de dados e da hipotese.
4. **Calcular Valor da Estatística:**
   * Usando a amostra, calcule o valor da estatística de teste.
5. **Determinar Valor Crítico:**
   * Valor que separa a região de aceitação de H0 da região de rejeição. É determinado usando a estatística de teste, o nível de significância e a distribuição amostral da estatística.
6. **Comparar Valor da Estatística e Valor Crítico:**
   * Se o valor da estatística for **maior** que o valor crítico, **rejeite** H0.
   * Se o valor da estatística for **menor** que o valor crítico, **não rejeite** H0.

**Fórmulas:**

**Média:**

* H0: μ = μ0
* Ha: μ ≠ μ0
* Estatística de Teste: z = (x̄ - μ0) / (σ/√n)

**Proporção:**

* H0: p = p0
* Ha: p ≠ p0
* Estatística de Teste: z = (p̂ - p0) / √(p0(1-p0)/n)

Onde:

* x̄ é a média da amostra
* σ é o desvio padrão da população (se conhecido) ou desvio padrão estimado (se desconhecido)
* n é o tamanho da amostra
* p̂ é a proporção da amostra
* μ0 e p0 são os valores especificados na hipótese nula

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

**Testes de Hipóteses: Testes de Comparação de Duas Amostras**

Os testes de comparação de duas amostras são usados para avaliar se as médias populacionais de dois grupos são diferentes. Os testes mais comuns são:

**Teste t para amostras independentes (dois lados):**

* H0: μ1 = μ2 (as médias são iguais)
* H1: μ1 ≠ μ2 (as médias são diferentes)

**Fórmula:**

t = (x̄1 - x̄2) / sqrt(s\_p² (1/n1 + 1/n2))

onde:

* x̄1 e x̄2 são as médias das amostras
* s\_p² é a variância agrupada, calculada como:
* s\_p² = ((n1 - 1)s1² + (n2 - 1)s2²) / (n1 + n2 -2)
* n1 e n2 são os tamanhos das amostras
* s1² e s2² são as variâncias das amostras

**Teste t para amostras pareadas (um lado):**

* H0: μd = 0 (as médias das diferenças são zero)
* H1: μd > 0 (as médias das diferenças são maiores que zero)

**Fórmula:**

t = x̄d / (sd / √n)

onde:

* x̄d é a média das diferenças entre as observações pareadas
* sd é o desvio padrão das diferenças
* n é o número de pares

**Teste de Wilcoxon-Mann-Whitney (não paramétrico):**

* H0: Não há diferença entre as distribuições das duas amostras
* H1: Existe uma diferença entre as distribuições das duas amostras

**Teste de sinais (não paramétrico):**

* H0: A mediana da diferença entre as observações pareadas é zero
* H1: A mediana da diferença entre as observações pareadas é diferente de zero

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

**Testes de Hipóteses: Teste de Normalidade (Qui-Quadrado)**

O teste de normalidade do qui-quadrado avalia se uma distribuição amostral segue uma distribuição normal.

**Procedimento:**

1. Divida os dados em intervalos de igual largura ou frequência.
2. Calcule a frequência observada em cada intervalo.
3. Calcule a frequência esperada em cada intervalo sob a suposição de normalidade.
4. Calcule o valor do qui-quadrado:

χ² = Σ [(O - E)² / E]

onde:

* O = frequência observada
* E = frequência esperada

**Hipóteses:**

* H0: A distribuição amostral segue uma distribuição normal.
* Ha: A distribuição amostral não segue uma distribuição normal.

**Rejeição da Hipótese Nula:**

Rejeite H0 se o valor do qui-quadrado for maior que o valor crítico encontrado na tabela de distribuição do qui-quadrado, com α graus de liberdade, onde α é o nível de significância.

**Limitações:**

* O teste é sensível ao tamanho da amostra.
* O teste pode ser afetado por outliers.
* O teste requer uma amostra de tamanho razoável (tipicamente maior que 50).

**Interpretação:**

* Se H0 for rejeitada, conclui-se que a distribuição amostral não segue uma distribuição normal.
* Se H0 não for rejeitada, não há evidências suficientes para concluir que a distribuição amostral é não normal.

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

**Testes de Hipóteses: Intervalos de Confiança**

**Objetivo:**

* Estimar um parâmetro populacional desconhecido com um certo nível de confiança.

**Procedimento:**

1. **Definir a hipótese nula (H0)**: Afirmação sobre o parâmetro a ser testada.
2. **Coletar uma amostra** da população.
3. **Calcular a estatística de teste** com base nos dados da amostra.
4. **Determinar o valor-p**: A probabilidade de observar a estatística de teste ou uma mais extrema, assumindo que H0 é verdadeira.
5. **Comparar o valor-p com o nível de significância (α)**: Se o valor-p for menor que α, rejeitar H0; caso contrário, não rejeitar H0.

**Intervalos de Confiança:**

Quando H0 não é rejeitada, é possível construir um intervalo de confiança para o parâmetro populacional.

**Intervalo de Confiança para uma Média Populacional (σ conhecido):**

μ ∈ (x̄ ± z\* \* σ/√n)

* **μ:** Média populacional
* **x̄:** Média amostral
* *z:*\* Valor crítico da distribuição normal com confiança de 1 - α
* **σ:** Desvio padrão conhecido da população
* **n:** Tamanho da amostra

**Intervalo de Confiança para uma Média Populacional (σ desconhecido):**

μ ∈ (x̄ ± t\* \* s/√n)

* *t:*\* Valor crítico da distribuição t de Student com graus de liberdade n-1 e confiança de 1 - α
* **s:** Desvio padrão amostral

**Vantagens:**

* Fornece uma estimativa do parâmetro com um nível de precisão conhecido.
* Permite que os pesquisadores façam inferências sobre a população a partir dos dados da amostra.

**Limitações:**

* Depende da representatividade da amostra.
* O nível de confiança não garante que o intervalo de confiança conterá o parâmetro verdadeiro.

Item do edital: Histogramas e curvas de frequência

**Histogramas**

Histrogramas são representações gráficas de distribuições de frequência, que mostram a frequência de ocorrência de valores de dados em intervalos distintos. São construídos:

* Dividindo o intervalo de dados em intervalos de largura igual (classes)
* Contando a frequência de dados em cada classe
* Representando a frequência como barras verticais

**Curvas de Frequência**

Curvas de frequência são representações gráficas suaves de distribuições de frequência. Elas são obtidas conectando os pontos médios das barras do histograma.

**Formas Comuns de Curvas de Frequência**

* **Normal (Gaussiana):** Curva em forma de sino simétrica com pico no centro.
* **Uniforme:** Curva plana, indicando que os valores de dados são igualmente prováveis de ocorrer.
* **Bimodal:** Curva com dois picos, indicando dois grupos distintos de dados.
* **Esqueva:** Curva assimétrica com cauda mais longa em um lado.

**Fórmulas para Curvas Normais**

A curva de frequência normal é definida pela equação:

f(x) = (1 / (σ√(2π))) \* e^(-(x - μ)² / (2σ²))

onde:

* x é o valor do dado
* μ é a média
* σ é o desvio padrão
* e é a base da exponencial natural (aproximadamente 2,71828)

**Utilização**

Histogramas e curvas de frequência são usados para:

* Visualizar distribuições de dados
* Identificar padrões e tendências
* Comparar distribuições
* Fazer inferências sobre populações de dados

Item do edital: Diagrama boxplot

**Diagrama Boxplot**

Um diagrama boxplot, também conhecido como diagrama de caixa e bigodes, é uma representação gráfica que exibe a distribuição de um conjunto de dados. Ele fornece uma visão geral resumida da distribuição, incluindo:

* **Mediana:** Linha que divide o diagrama ao meio, separando a metade superior da metade inferior dos dados.
* **Média:** Símbolo (geralmente um círculo ou diamante) que indica o valor médio dos dados.
* **Quartis (Q1, Q3):** Linhas que dividem a metade superior e inferior dos dados ao meio.
* **Intervalo Interquartil (IQR):** A diferença entre Q3 e Q1, que mede a variabilidade dos dados no meio 50%.
* **Bigodes:** Linhas que se estendem de Q1 e Q3 para baixo e para cima, respectivamente, até o valor máximo e mínimo que não são considerados outliers.
* **Outliers:** Pontos que ficam fora dos bigodes e são considerados valores extremos.

**Fórmulas:**

* Mediana = (N+1)/2
* Quartis (Q1, Q3) = (N+1)/4
* Intervalo Interquartil (IQR) = Q3 - Q1

**Interpretação:**

* O tamanho da caixa representa a variabilidade dos dados no meio 50%.
* A posição da mediana indica o valor central dos dados.
* Os bigodes mostram a extensão da distribuição e a presença de valores extremos.
* Os outliers são pontos que devem ser investigados mais detalhadamente.

**Vantagens:**

* Fornece uma visão geral rápida e abrangente da distribuição dos dados.
* Pode identificar outliers e distorções na distribuição.
* Permite comparações fáceis entre diferentes conjuntos de dados.

**Limitações:**

* Não mostra a forma detalhada da distribuição.
* Pode ser difícil interpretar quando há muitos outliers.
* Não é adequado para conjuntos de dados com distribuições bimodais ou multimodais.

Item do edital: Avaliação de outliers.

**Avaliação de Outliers**

Os outliers são dados que se desviam significativamente do resto dos dados em um conjunto de dados. Identificá-los é crucial para garantir a precisão e confiabilidade das análises. Aqui estão os métodos comuns:

**Métodos Gráficos:**

* **Diagramas de dispersão:** Visualize os dados e identifique pontos que se desviam da tendência geral.
* **Box plots:** Esses gráficos representam a distribuição de dados com quartis (Q1, Q2, Q3) e valores extremos. Os outliers são pontos que estão fora dos quartis (~1,5 x IQR).

**Métodos Estatísticos:**

* **Teste de Chauvenet:** Calcula a probabilidade de um dado ser um outlier e remove dados com probabilidades baixas (geralmente < 0,05).
* **Teste de Grubbs:** Usado para conjuntos de dados pequenos (N < 100). Calcula o valor z para o dado suspeito e o compara com o valor crítico para um nível de significância específico.
* **Teste de Dixon:** Semelhante ao teste de Grubbs, mas usado para conjuntos de dados maiores (N > 100).

**Medidas de Outliers:**

* **Desvio Absoluto Médio (MAD):** A mediana das distâncias absolutas entre os dados e o seu valor mediano.
* **Intervalo Interquartil (IQR):** A diferença entre o terceiro e o primeiro quartil.
* **Fórmula de Fences:** Intervalo entre Q1 - 1,5 \* IQR e Q3 + 1,5 \* IQR. Dados fora deste intervalo são considerados outliers.

**Remoção de Outliers:**

A decisão de remover outliers deve ser tomada com cuidado, pois eles podem conter informações valiosas. A remoção só é recomendada quando os outliers são erros de medição ou pontos de dados genuinamente atípicos que não representam a população geral.

**Conclusão:**

A avaliação de outliers é essencial para a integridade dos dados. Ao usar métodos gráficos e estatísticos, os analistas podem identificar e lidar com outliers, garantindo que as análises e interpretações sejam precisas e confiáveis.

Item do edital: Técnicas de classificação: Naive Bayes

**Técnicas de Classificação: Naive Bayes**

Naive Bayes é uma técnica de classificação probabilística que se baseia no Teorema de Bayes. É um método simples e eficiente que pode ser usado para uma ampla variedade de problemas de classificação.

**Princípios:**

* Naive Bayes assume que os recursos da instância são condicionalmente independentes dadas a classe. Embora essa suposição nem sempre seja verdadeira, ela permite que o modelo seja computado eficientemente.
* O modelo calcula a probabilidade posterior de cada classe, dada a instância, e atribui a classe com a probabilidade posterior mais alta.

**Fórmula:**

A probabilidade posterior de uma classe *c* dada uma instância *x* é calculada como:

P(c | x) = P(x | c) \* P(c) / P(x)

onde:

* P(x | c) é a probabilidade de observar a instância *x* dada a classe *c* (likelihood)
* P(c) é a probabilidade a priori da classe *c* (prior)
* P(x) é a probabilidade da instância *x* (evidência), que é constante para todas as classes

**Vantagens:**

* Simples e fácil de implementar
* Eficiente computacionalmente
* Pode lidar com dados de alta dimensão
* Robusto ao ruído e outliers

**Desvantagens:**

* A suposição de independência condicional pode não ser realista em alguns casos
* Pode ser afetado por recursos redundantes ou irrelevantes

**Aplicações:**

Naive Bayes é amplamente utilizado em vários domínios, incluindo:

* Classificação de texto
* Detecção de spam
* Análise de sentimento
* Diagnóstico médico

Item do edital: Técnica de classificação Regressão logística

**Técnica de Classificação por Regressão Logística**

A regressão logística é uma técnica de classificação estatística usada para prever a probabilidade de um evento binário (ou seja, com dois resultados possíveis). É uma extensão da regressão linear que usa uma função logística para modelar a probabilidade do evento.

**Fórmula:**

A função logística é definida como:

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

onde:

* p é a probabilidade do evento
* z é uma combinação linear de variáveis ​​preditoras (x) e seus respectivos coeficientes (β):

z = β0 + β1x1 + β2x2 + ... + βnxn

**Procedimento:**

1. **Coletar dados:** Colete um conjunto de dados que contenha as variáveis ​​preditoras e a variável de resposta binária.
2. **Estimar coeficientes:** Use métodos de otimização (por exemplo, máxima verossimilhança) para estimar os coeficientes β da função logística.
3. **Interpretar coeficientes:** Os coeficientes β indicam a influência das variáveis ​​preditoras na probabilidade do evento. Um coeficiente positivo indica uma relação positiva, enquanto um coeficiente negativo indica uma relação negativa.
4. **Prever probabilidades:** Uma vez que os coeficientes são estimados, a função logística pode ser usada para prever a probabilidade do evento para novas observações.
5. **Classificar observações:** As observações podem ser classificadas como pertencentes a uma das duas classes (ou seja, o evento ocorreu ou não) com base em um limiar de probabilidade (por exemplo, 0,5).

**Vantagens:**

* Maneja variáveis ​​preditoras contínuas e categóricas
* Fornece probabilidades de eventos
* Pode ser interpretado facilmente
* Robusto contra outliers

**Desvantagens:**

* Assumir uma relação linear entre a probabilidade do evento e as variáveis ​​preditoras
* Pode ser sensível a desequilíbrios de classes
* Requer um tamanho de amostra relativamente grande

Item do edital: Técnica de classificação Redes neurais artificiais

**Técnicas de Classificação com Redes Neurais Artificiais (RNAs)**

As RNAs são modelos de aprendizado de máquina inspirados na estrutura e função do cérebro humano. Elas podem ser usadas para classificar dados em categorias pré-definidas.

**Principais Abordagens de Classificação:**

* **Classificação Binária:** Classifica dados em duas categorias (por exemplo, positivo/negativo).
* **Classificação Multivariada:** Classifica dados em mais de duas categorias (por exemplo, dígitos de 0 a 9).
* **Classificação Hierárquica:** Divide os dados em uma hierarquia de classes (por exemplo, reino, filo, classe, ordem).

**Modelos de RNA para Classificação:**

* **Perceptron Multicamadas (MLP):** Um modelo feedforward com múltiplas camadas de nós ocultos.
* **Redes Neurais Convolucionais (CNNs):** Especializadas em processar dados de imagem, usando operações de convolução.
* **Redes Neurais Recorrentes (RNNs):** Projetadas para manipular dados sequenciais, como texto e dados de séries temporais.

**Fórmulas:**

* **Função de Ativação:** Determina a saída de um neurônio. Exemplos incluem sigmoid, tanh e ReLU.
* **Função de Perda:** Mede a diferença entre as previsões da RNA e os rótulos verdadeiros. Exemplos incluem perda de entropia cruzada e perda de erro quadrático médio.
* **Propagação para Trás:** Algoritmo usado para calcular os gradientes da função de perda em relação aos pesos da RNA.

**Etapas da Classificação:**

* **Pré-processamento de Dados:** Preparação dos dados para treinamento, incluindo limpeza, normalização e codificação.
* **Treinamento da RNA:** Otimização dos pesos da RNA para minimizar a função de perda usando dados de treinamento rotulados.
* **Avaliação:** Avaliação do desempenho da RNA em dados de teste independentes.
* **Implantação:** Uso da RNA treinada para classificar novos dados não vistos.

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

**Árvores de Decisão**

As Árvores de Decisão são uma técnica de classificação supervisionada baseada em dividir iterativamente um conjunto de dados em subconjuntos menores. Cada nó na árvore representa um atributo ou característica, enquanto cada ramificação representa um possível valor desse atributo. As folhas da árvore representam as classes ou previsões esperadas.

**Algoritmo ID3**

O algoritmo ID3 (Iterative Dichotomiser 3) é um método de construção de árvores de decisão que usa a Entropia de Informação para selecionar o atributo que melhor divide os dados.

**Fórmula para Cálculo da Entropia:**

H(S) = - Σ (p\_i log\_2 p\_i)

Onde:

* H(S) é a Entropia do conjunto de dados S
* p\_i é a probabilidade da classe i

**Algoritmo C4.5**

O algoritmo C4.5 (Successor of ID3) é uma extensão do ID3 que usa Gain Ratio para selecionar atributos. O Gain Ratio é uma medida que considera a relação entre a Informação de Ganho e a Entropia de Dados Dividida.

**Fórmula para Cálculo do Gain Ratio:**

GR(S, A) = (H(S) - H(S, A)) / H(S, A)

Onde:

* GR(S, A) é o Ganho Ratio do atributo A no conjunto de dados S
* H(S) é a Entropia de S
* H(S, A) é a Entropia de S após a divisão pelo atributo A

**Vantagens das Árvores de Decisão:**

* Fáceis de entender e interpretar
* Robustas a dados ausentes e barulhentos
* Podem lidar com dados categóricos e numéricos
* Não requerem pré-processamento extensivo

**Desvantagens das Árvores de Decisão:**

* Podem ser propensas a sobreajuste se não forem podadas corretamente
* Podem se tornar muito profundas e complexas com grandes conjuntos de dados
* A ordem dos atributos na construção da árvore pode afetar o resultado

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

**Técnica de Classificação Florestas Aleatórias (Random Forest)**

**Conceito:** Florestas aleatórias são um método de conjunto de aprendizado de máquina que cria uma coleção de árvores de decisão individuais para melhorar a precisão da classificação. Cada árvore é treinada em um subconjunto aleatório dos dados e com um subconjunto aleatório de recursos.

**Algoritmo:**

1. **Selecione aleatoriamente uma amostra:** De n amostras de dados, selecione uma amostra inicial de m amostras para cada árvore no conjunto.
2. **Construa uma árvore de decisão:** Construa uma árvore de decisão para cada amostra inicial usando um subconjunto aleatório de p recursos.
3. **Repita o Passo 1-2:** Repita os Passos 1 e 2 para criar um número predeterminado de árvores.
4. **Classifique:** Para classificar uma nova entrada, execute-a por cada árvore no conjunto e gere uma classificação de maioria.

**Fórmulas:**

* **Precisão da árvore individual:**

Precision = (True Positives) / (True Positives + False Positives)

* **Precisão da floresta aleatória:**

Precision\_RF = (TP + TN) / (TP + FP + FN + TN)

onde:

* TP = Verdadeiros Positivos
* FP = Falsos Positivos
* FN = Falsos Negativos
* TN = Verdadeiros Negativos

**Vantagens:**

* Alta precisão
* Robusto a ruído e valores ausentes
* Pode lidar com dados de alta dimensão
* Importância das variáveis pode ser medida

**Desvantagens:**

* Pode ser computacionalmente caro
* Pode sofrer com superajuste se o número de árvores for muito grande
* Difícil de interpretar os resultados com muitas árvores

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

**Técnica de Classificação Máquinas de Vetores de Suporte (SVM)**

As máquinas de vetores de suporte (SVM) são algoritmos de classificação supervisionados usados para resolver problemas de classificação binária e múltipla. São baseados na ideia de mapear dados não lineares em um espaço de dimensão superior, onde se torna linearmente separável.

**Conceitos Básicos:**

* **Vetor de Suporte:** Um ponto de dado que determina a margem de classificação.
* **Margem:** A distância entre os vetores de suporte e o hiperplano de decisão.
* **Hiperplano de Decisão:** Uma linha ou plano no espaço de dimensão superior que separa as classes.

**Função Objetivo:**

As SVM visam maximizar a margem entre os vetores de suporte e o hiperplano de decisão. A função objetivo é:

max w'w

sujeito a: y\_i (w'x\_i + b) >= 1, para todo i

onde:

* w é o vetor de pesos
* b é o bias
* x\_i é o i-ésimo ponto de dados
* y\_i é a etiqueta de classe do i-ésimo ponto de dados

**Algoritmo:**

1. Mapear os dados não lineares para um espaço de dimensão superior usando um kernel.
2. Resolver a função objetivo usando programação quadrática.
3. Obter o hiperplano de decisão e os vetores de suporte.

**Vantagens:**

* Altamente eficazes em conjuntos de dados de alta dimensão.
* Robustas a ruídos e sobreajuste.
* Podem lidar com dados não lineares por meio de kernels.

**Desvantagens:**

* Pode ser computacionalmente intensivo para conjuntos de dados grandes.
* A seleção do kernel e dos parâmetros pode ser desafiadora.

**Aplicações:**

As SVM são amplamente utilizadas em diversas aplicações, incluindo:

* Classificação de imagens
* Reconhecimento de fala
* Análise de texto
* Bioinformática

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

**Técnica de Classificação K Vizinhos Mais Próximos (KNN)**

A KNN é um algoritmo de aprendizado supervisionado que classifica novos pontos de dados com base na proximidade com os pontos de dados de treinamento rotulados.

**Como funciona a KNN:**

1. **Selecionar K:** Escolha um valor K, representando o número de vizinhos mais próximos a serem considerados.
2. **Calcular Distâncias:** Calcule a distância entre o novo ponto de dados e todos os pontos de dados de treinamento.
3. **Identificar Vizinhos Mais Próximos:** Identifique os K pontos de dados de treinamento mais próximos do novo ponto de dados.
4. **Classificar:** Determine a classe predominante entre os K vizinhos mais próximos. O novo ponto de dados é atribuído à classe mais frequente.

**Fórmulas:**

* **Distância Euclidiana:** d(x, y) = sqrt((x1 - y1)^2 + (x2 - y2)^2 + ...) para pontos em espaço n-dimensional.
* **Distância de Manhattan:** d(x, y) = |x1 - y1| + |x2 - y2| + ... para pontos em espaço n-dimensional.

**Vantagens da KNN:**

* Simples e fácil de implementar.
* Não requer treinamento, tornando-o rápido.
* Pode lidar com dados com muitas dimensões.

**Desvantagens da KNN:**

* Pode ser sensível à escolha de K.
* Pode ser lento em conjuntos de dados grandes.
* Não pode identificar relacionamentos complexos entre recursos.

**Aplicações da KNN:**

* Classificação de texto
* Reconhecimento de padrões
* Previsão financeira

Item do edital: Técnica de classificação

**Técnicas de Classificação**

As técnicas de classificação visam organizar um conjunto de dados em grupos distintos e significativos. Elas são usadas em diversas áreas, como mineração de dados, reconhecimento de padrões e tomada de decisão.

**Tipos de Algoritmos de Classificação**

* **Árvore de Decisão:** Divide recursivamente os dados em grupos com base em critérios específicos.
* **Floresta Aleatória:** Cria vários modelos de árvore de decisão e combina seus resultados.
* **Rede Neuronal:** Modelo computacional inspirado no cérebro humano que pode aprender com os dados.
* **Máquina de Vetores de Suporte (SVM):** Usa linhas ou hiperplanos para separar os dados em classes.
* **k-Vizinhos Mais Próximos (k-NN):** Classifica pontos de dados com base na similaridade com seus k vizinhos mais próximos.

**Fórmulas Comuns**

* **Entropia (H):** Mede a incerteza em uma distribuição de probabilidade:

H(p) = -∑[p(x) \* log2(p(x))]

* **Ganho de Informação (IG):** Mede a redução da incerteza após dividir os dados em subgrupos:

IG(X,Y) = H(Y) - H(Y | X)

* **Coeficiente Kappa:** Mede a concordância entre as classificações previstas e reais:

Kappa = (P(A) - P(E)) / (1 - P(E))

**Características a Considerar**

* **Tipo de Variáveis:** Categóricas ou numéricas
* **Número de Classes:** Binário ou multiclasse
* **Tamanho do Conjunto de Dados:** Pequeno, médio ou grande
* **Desempenho:** Precisão, revocação e pontuação F1

**Usos**

* Identificação de padrões em dados
* Previsão de classes ou resultados
* Agrupamento de clientes
* Detecção de fraudes
* Classificação de imagens

Item do edital: Avaliação de modelos de classificação: treinamento

**Avaliação de Modelos de Classificação: Treinamento**

O treinamento é uma etapa crucial na avaliação de modelos de classificação, pois ajuda a determinar o desempenho e a capacidade de generalização do modelo.

**Avaliação do Desempenho do Treinamento**

* **Precisão:** Porcentagem de previsões corretas.
* **Recall:** Porcentagem de instâncias positivas corretamente previstas.
* **Especificidade:** Porcentagem de instâncias negativas corretamente previstas.
* **Curva ROC (Receiver Operating Characteristic):** Curva que representa a variação da taxa de verdadeiros positivos (TPR) em relação à taxa de falsos positivos (FPR) em diferentes limiares de decisão.

**Métricas de Regularização**

* **Penalidade L1 (LASSO):** λ∑j|wj|
* **Penalidade L2 (Ridge):** λ∑j(wj)^2

**Avaliação da Capacidade de Generalização**

A capacidade de um modelo de generalizar além dos dados de treinamento é vital. Métricas como a validação cruzada e o conjunto de teste de holdout são usadas para avaliar a generalização.

**Validação Cruzada**

* Divide os dados de treinamento em várias dobras.
* Treina e avalia o modelo em cada dobra.
* A pontuação de desempenho é a média das pontuações de todas as dobras.

**Conjunto de Teste de Holdout**

* Separa uma parte dos dados de treinamento como um conjunto de teste.
* O modelo é treinado nos dados restantes e avaliado no conjunto de teste.

**Fórmulas**

* **Precisão:** Precisão = TP / (TP + FP)
* **Recall:** Recall = TP / (TP + FN)
* **Especificidade:** Especificidade = TN / (TN + FP)

Item do edital: Avaliação de modelos de classificação: teste

**Avaliação de Modelos de Classificação: Teste**

**Objetivo:**

O teste avalia o desempenho do modelo de classificação em dados não vistos durante o treinamento.

**Procedimento:**

1. **Divisão dos Dados:** Divida o conjunto de dados em conjuntos de treinamento e teste disjuntos.
2. **Treinamento do Modelo:** Treine o modelo no conjunto de treinamento.
3. **Previsão do Teste:** Use o modelo treinado para prever as classes dos dados de teste.
4. **Avaliação das Métricas:** Calcule as métricas de avaliação no conjunto de teste.

**Métricas de Avaliação Comuns:**

* **Precisão:** Propensão do modelo de prever corretamente as classes.
  + Fórmula: Precisão = Verdadeiros Positivos / (Verdadeiros Positivos + Falsos Positivos)
* **Rechamada:** Capacidade do modelo de identificar corretamente as instâncias positivas.
  + Fórmula: Rechamada = Verdadeiros Positivos / (Verdadeiros Positivos + Falsos Negativos)
* **F1-score:** Média harmônica de precisão e recall.
  + Fórmula: F1-score = (2 \* Precisão \* Rechamada) / (Precisão + Rechamada)
* **Área sob a Curva ROC (AUC-ROC):** Área sob a curva Receiver Operating Characteristic (ROC), que mede a capacidade do modelo de distinguir entre classes.
* **Área sob a Curva PR (AUC-PR):** Área sob a curva Precision-Recall, que mede a capacidade do modelo de recuperar instâncias positivas.
* **Curva ROC:** Plot da Taxa de Verdadeiro Positivo (TPR) versus a Taxa de Falso Positivo (FPR) em diferentes limiares de decisão.

**Interpretação:**

As métricas de avaliação fornecem insights sobre o desempenho do modelo em novos dados. Um modelo com alta precisão e recall indica boa capacidade de previsão. Uma AUC-ROC alta indica boa capacidade de discriminação, enquanto uma AUC-PR alta indica boa capacidade de recuperação.

**Limitações:**

* O teste depende da representatividade do conjunto de teste.
* As métricas de avaliação podem variar dependendo do contexto e do objetivo da classificação.

Item do edital: Avaliação de modelos de classificação: validação

**Avaliação de Modelos de Classificação: Validação**

A validação é um passo crucial na avaliação de modelos de classificação, garantindo que o modelo seja generalizável para novos dados e avaliando seu desempenho em situações do mundo real.

**Métodos de Validação:**

Existem vários métodos de validação comumente usados:

* **Validação Cruzada:** Divide o conjunto de dados em subconjuntos menores (dobras) e treina o modelo usando cada dobra como conjunto de validação.
* **Validação de Subconjunto de Validação:** Divide o conjunto de dados em três subconjuntos: treinamento, validação e teste. O modelo é treinado no conjunto de treinamento, ajustado no conjunto de validação e avaliado no conjunto de teste.
* **Amostragem de Inicialização:** Treina o modelo em diferentes subconjuntos aleatórios do conjunto de dados original e agrega os resultados.

**Métricas de Desempenho:**

As métricas de desempenho comuns usadas na validação de modelos de classificação incluem:

* **Precisão:** Número de previsões corretas dividido pelo número de todas as previsões.
* **Recall:** Número de previsões corretas da classe positiva dividido pelo número real de instâncias da classe positiva.
* **F1-Score:** Média harmônica de precisão e recall.

**Fórmula F1-Score:**

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

**Objetivo da Validação:**

O objetivo da validação é estimar o erro do modelo em dados novos e evitar o overfitting, que ocorre quando o modelo se ajusta muito aos dados de treinamento específicos.

**Conclusão:**

A validação é essencial para avaliar o desempenho e a generalização dos modelos de classificação. Ao usar métodos de validação e métricas de desempenho adequados, os cientistas de dados podem obter uma compreensão abrangente do desempenho do modelo e tomar decisões informadas sobre sua implantação.

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

**Avaliação de Modelos de Classificação: Validação Cruzada**

A validação cruzada é um método estatístico usado para avaliar o desempenho de modelos de classificação. Ele divide os dados disponíveis em subconjuntos (dobras) e treina o modelo iterativamente em diferentes combinações dessas dobras.

**Funcionamento:**

1. Divida os dados em **k** dobras de tamanho aproximadamente igual.
2. Para cada dobra **i**:
   * Treine o modelo nas **k-1** dobras restantes.
   * Avalie o modelo na dobra **i**.
3. Calcule a métrica de avaliação (por exemplo, precisão, recall) para cada dobra.
4. Retorne a média ou mediana das métricas de avaliação para todas as dobras.

**Fórmulas:**

* Precisão: TP / (TP + FP)
* Recall: TP / (TP + FN)
* Valor-F1: 2 \* (Precisão \* Recall) / (Precisão + Recall)

**Tipos de Validação Cruzada:**

* **Validação Cruzada k-dobras:** Divide os dados em **k** dobras iguais e treina o modelo em **k-1** dobras por iteração.
* **Validação Cruzada de Deixe um de Fora:** Um caso especial de validação cruzada k-dobras com **k = n**, onde **n** é o número de observações nos dados.
* **Validação Cruzada Estratificada:** Garante que as dobras tenham a mesma proporção de classes de rótulo que os dados originais.

**Vantagens:**

* Reduz o viés de partição de dados.
* Fornece uma estimativa mais confiável do desempenho do modelo.
* Permite a avaliação de vários modelos usando o mesmo conjunto de dados.

**Desvantagens:**

* Pode ser computacionalmente caro, especialmente para conjuntos de dados grandes.
* Os resultados podem variar dependendo do número de dobras e da ordem em que as dobras são usadas.

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

**Matriz de Confusão**

A matriz de confusão é uma métrica de avaliação comumente usada para classificação. Ela representa o desempenho de um modelo de classificação por meio da contagem de predições verdadeiras e falsas para diferentes classes.

**Estrutura:**

| Verdadeiro Positivo (TP) | Falso Negativo (FN) | |---|---|---| | Falso Positivo (FP) | Verdadeiro Negativo (TN) |

**Fórmulas:**

* **Precisão:** TP / (TP + FP)
* **Revocação:** TP / (TP + FN)
* **Escore F1:** 2 \* Precisão \* Revocação / (Precisão + Revocação)

**Interpretação:**

* **Alta TP e TN:** O modelo está classificando bem.
* **Alto FP:** O modelo está fazendo predições falsas positivas.
* **Alto FN:** O modelo está perdendo predições verdadeiras positivas.

**Vantagens:**

* Fornece uma visão abrangente do desempenho do modelo.
* Permite calcular métricas adicionais, como precisão, revocação e escore F1.

**Limitações:**

* Pode ser tendenciosa para conjuntos de dados desbalanceados.
* Não considera a ordem das predições.

Item do edital: Avaliação de modelos de classificação: acurácia

**Avaliação de Acurácia na Classificação**

A acurácia é uma métrica fundamental para avaliar o desempenho de modelos de classificação, medindo a proporção de previsões corretas feitas pelo modelo.

**Definição:**

Acurácia = (Número de previsões corretas) / (Número total de previsões)

**Interpretação:**

* Uma acurácia de 1,0 indica que o modelo previu corretamente todas as instâncias.
* Uma acurácia de 0,0 indica que o modelo não previu corretamente nenhuma instância.
* Uma acurácia de 0,5 indica que o modelo está adivinhando aleatoriamente.

**Vantagens:**

* Métrica intuitiva e fácil de entender.
* Pode ser aplicada a qualquer conjunto de dados de classificação.

**Desvantagens:**

* Pode ser tendencioso para conjuntos de dados desequilibrados, onde uma classe domina as outras.
* Não considera a distribuição de classes prevista.
* Pode não ser informativo para modelos que produzem previsões probabilísticas.

**Fórmulas Relacionadas:**

* **Falso Negativo (FN):** Número de instâncias que foram classificadas incorretamente como negativas, mas deveriam ter sido positivas.
* **Falso Positivo (FP):** Número de instâncias que foram classificadas incorretamente como positivas, mas deveriam ter sido negativas.
* **Verdadeiro Positivo (TP):** Número de instâncias que foram classificadas corretamente como positivas.
* **Verdadeiro Negativo (TN):** Número de instâncias que foram classificadas corretamente como negativas.

**Cálculo da Acurácia:**

Acurácia = (TP + TN) / (TP + TN + FP + FN)

Item do edital: Avaliação de modelos de classificação: precisão

**Avaliação de Classificação: Precisão**

A precisão, também conhecida como Valor Preditivo Positivo (PPV), é uma métrica de avaliação que mede a proporção de previsões positivas corretas para o número total de previsões positivas feitas. É calculada como:

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

**Interpretação:**

* Um valor de precisão alto indica que o modelo é bom em identificar corretamente os exemplos positivos.
* Um valor de precisão baixo indica que o modelo pode estar fazendo muitas previsões positivas falsas.

**Limitações:**

* A precisão pode ser enviesada quando há uma distribuição desequilibrada de classes (por exemplo, mais amostras positivas do que negativas).
* A precisão não considera os falsos negativos (previsões positivas incorretas).

**Fórmulas Relacionadas:**

* **Recall (Sensibilidade):** Verdadeiros Positivos / (Verdadeiros Positivos + Falsos Negativos)
* **Pontuação F1:** 2 \* (Precisão \* Recall) / (Precisão + Recall)
* **Curva ROC:** Um gráfico que mostra o trade-off entre Falsos Positivos e Verdadeiros Positivos

Item do edital: Avaliação de modelos de classificação: revocação

**Avaliação de Modelos de Classificação: Revocação**

A revocação, também conhecida como sensibilidade, é uma métrica que mede a capacidade de um modelo de classificação em identificar corretamente as instâncias positivas (verdadeiros positivos) do total de instâncias positivas verdadeiras. É calculada como:

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

**Interpretação:**

* Um valor de revocação alto indica que o modelo é bom em identificar corretamente as instâncias positivas.
* Um valor de revocação baixo indica que o modelo está perdendo muitas instâncias positivas verdadeiras.

**Vantagens:**

* Mede a capacidade do modelo de evitar falsos negativos (instâncias positivas classificadas como negativas).
* É útil quando é crucial identificar todas as instâncias positivas verdadeiras.

**Desvantagens:**

* Pode ser afetado por desequilíbrios de classes, onde uma classe tem muito mais instâncias que outra.
* Não leva em consideração os falsos positivos (instâncias negativas classificadas como positivas).

**Considerações de uso:**

* A revocação é mais adequada para problemas de classificação onde falsos negativos são caros ou indesejáveis.
* Deve ser usado em conjunto com outras métricas, como precisão e F1-score, para uma avaliação abrangente do modelo.

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

**Avaliação de Modelos de Classificação: F1-Score**

**Introdução:** O F1-score é uma métrica de desempenho amplamente utilizada para avaliar modelos de classificação binária, que mede tanto a precisão quanto a revocação.

**Fórmula:**

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

**Interpretação:**

* Um F1-score de 1 indica que o modelo classifica perfeitamente todos os exemplos.
* Um F1-score de 0 indica que o modelo não classifica corretamente nenhum exemplo.

**Como interpretar o F1-score:** O F1-score considera o equilíbrio entre Precisão e Revocação:

* **Precisão** mede a proporção de exemplos classificados como positivos que são realmente positivos.
* **Revocação** mede a proporção de exemplos positivos que são classificados corretamente pelo modelo.

Um alto F1-score indica que o modelo classifica com sucesso os exemplos positivos e negativos.

**Vantagens:**

* Considera Precisão e Revocação, fornecendo uma métrica equilibrada.
* Pode ser usado em problemas de classificação binária.
* Simples de entender e interpretar.

**Desvantagens:**

* Pode ser sensível a dados desbalanceados, onde uma classe é significativamente mais frequente que a outra.
* Não considera a sensibilidade e especificidade, que são métricas importantes em alguns domínios.

**Conclusão:** O F1-score é uma métrica útil para avaliar o desempenho de modelos de classificação binária. Ao equilibrar Precisão e Revocação, fornece uma visão geral da capacidade do modelo de classificar corretamente exemplos positivos e negativos.

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

**Avaliação de Modelos de Classificação: Curva ROC**

A curva ROC (Receiver Operating Characteristic) é uma medida de desempenho amplamente usada para avaliar modelos de classificação. Ela plota a taxa de verdadeiros positivos (TPR) contra a taxa de falsos positivos (FPR) em todos os limiares de classificação possíveis.

**Construindo uma Curva ROC:**

1. Ordene as instâncias do conjunto de dados por sua pontuação de classificação decrescente.
2. Para cada instância, calcule TPR e FPR como:

TPR = TP / (TP + FN)

FPR = FP / (FP + TN)

onde:

* TP: Verdadeiros positivos (instâncias positivas corretamente classificadas)
* FN: Falsos negativos (instâncias positivas incorretamente classificadas)
* FP: Falsos positivos (instâncias negativas incorretamente classificadas)
* TN: Verdadeiros negativos (instâncias negativas corretamente classificadas)

1. Plote TPR no eixo y contra FPR no eixo x.

**Interpretação da Curva ROC:**

* **Curvas Perfeitas:** Uma curva perfeita é uma linha diagonal de (0,0) a (1,1), indicando classificação perfeita.
* **Curvas Aleatórias:** Uma curva aleatória é uma linha horizontal em TPR = 0,5, indicando classificação aleatória.
* **Melhores Curvas:** Curvas acima da diagonal são melhores que linhas aleatórias, indicando melhor classificação.
* **Área sob a Curva (AUC):** Medida resumida do desempenho, representando a probabilidade do modelo classificar corretamente uma instância positiva e uma instância negativa.

**Vantagens da Curva ROC:**

* **Independente do Limiar:** Avalia o desempenho em todos os limiares, permitindo ajustes para diferentes cenários de decisão.
* **Robusta a Desequilíbrios de Classe:** Pode lidar com conjuntos de dados desequilibrados, onde uma classe é muito mais prevalente que a outra.
* **Comparação de Modelos:** Permite comparar facilmente o desempenho de diferentes modelos de classificação.

**Fórmulas Adicionais:**

* **Sensibilidade (Recall):** TPR
* **Especificidade:** 1 - FPR
* **Precisão:** TP / (TP + FP)

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

**Técnicas de Regressão: Redes Neurais para Regressão**

As Redes Neurais (RNs) são uma família de algoritmos de aprendizado de máquina inspirados no funcionamento do cérebro humano. São especialmente eficazes em tarefas que envolvem regressão, onde o objetivo é prever valores contínuos.

**Arquitetura das Redes Neurais para Regressão**

* **Camada de Entrada:** Recebe os dados de entrada.
* **Camadas Ocultas:** Pode haver uma ou mais camadas ocultas que processam os dados de entrada por meio de funções não lineares.
* **Camada de Saída:** Prediz o valor contínuo.

Cada camada consiste em nós (neurônios) interconectados por pesos. Os pesos são ajustados durante o processo de treinamento para minimizar o erro de previsão.

**Função de Perda**

A função de perda mede o erro entre as previsões e os valores reais. Uma função de perda comum para regressão é o erro quadrático médio (MSE):

MSE = (1/n) Σ(y\_i - f(x\_i))^2

Onde:

* n é o número de pontos de dados
* y\_i é o valor real
* f(x\_i) é a previsão da rede neural

**Algoritmo de Treinamento**

As redes neurais são treinadas usando o algoritmo de retropropagação. Este algoritmo ajusta iterativamente os pesos para minimizar a função de perda:

1. Feedforward: Os dados são propagados através da rede neural e uma previsão é feita.
2. Backpropagation: O erro é calculado e propagado para trás através da rede.
3. Atualização de pesos: Os pesos são ajustados na direção oposta ao gradiente da função de perda.

**Vantagens das Redes Neurais para Regressão**

* **Não linearidade:** Pode lidar com dados não lineares.
* **Alta capacidade:** Pode modelar funções complexas.
* **Generalização:** Pode generalizar bem para novos dados não vistos.

**Desvantagens das Redes Neurais para Regressão**

* **Excesso de ajuste (overfitting):** Podem aprender muito bem com os dados de treinamento, mas não generalizar bem para novos dados.
* **Caixa preta:** O funcionamento interno pode ser difícil de interpretar.
* **Tempo de treinamento:** Pode levar muito tempo para treinar redes neurais complexas.

Item do edital: Árvores de decisão para regressão

**Árvores de Decisão para Regressão**

As árvores de decisão são uma técnica de aprendizado de máquina usada para prever valores numéricos (também conhecidos como valores alvo ou variáveis dependentes) com base em um conjunto de variáveis preditivas (também conhecidas como variáveis independentes ou recursos).

**Funcionamento:**

Uma árvore de decisão para regressão é construída dividindo recursivamente o conjunto de dados em subconjuntos menores com base nos valores das variáveis preditivas. Cada nó interno representa uma variável preditiva e cada ramo representa um valor possível para essa variável. As folhas representam as previsões para os valores alvo.

**Divisão de Dados:**

Em cada nó, o conjunto de dados é dividido usando uma métrica de qualidade da divisão, como ganho de informação ou redução de variância. A métrica seleciona a variável preditiva e o valor de divisão que criam os subconjuntos mais homogêneos em termos dos valores alvo.

**Recursão:**

O processo de divisão é repetido recursivamente até que um dos seguintes critérios de parada seja atendido:

* O número máximo de divisões é atingido.
* Os dados restantes em um nó são muito pequenos.
* Os valores alvo em um nó são muito homogêneos.

**Previsão:**

Para fazer uma previsão para um novo dado, ele é percorrido pela árvore de decisão começando no nó raiz. Em cada nó, o valor correspondente à variável preditiva no novo dado é usado para navegar para o ramo apropriado. O valor alvo é previsto na folha atingida.

**Fórmulas:**

**Ganho de Informação:**

Ganho de Informação = H(Y) - H(Y|X)

onde:

* H(Y) é a entropia dos valores alvo antes da divisão
* H(Y|X) é a entropia condicional dos valores alvo após a divisão pela variável preditiva X

**Redução de Variância:**

Redução de Variância = Var(Y) - Var(Y|X)

onde:

* Var(Y) é a variância dos valores alvo antes da divisão
* Var(Y|X) é a variância condicional dos valores alvo após a divisão pela variável preditiva X

Item do edital: Máquinas de vetores de suporte para regressão

**Máquinas de Vetores de Suporte para Regressão (SVR)**

As Máquinas de Vetores de Suporte (SVM) são um algoritmo de aprendizado de máquina supervisionado originalmente desenvolvido para tarefas de classificação. Entretanto, elas também podem ser adaptadas para tarefas de regressão, conhecidas como Máquinas de Vetores de Suporte para Regressão (SVR).

**Objetivo:**

O objetivo das SVR é encontrar uma função que minimize o erro de previsão do valor-alvo enquanto mantém bons limites de generalização.

**Funcionamento:**

As SVR funcionam mapeando os dados de entrada para um espaço de características de dimensão superior usando uma função de kernel. Uma função de perda de insensibilidade é então aplicada aos erros de previsão, permitindo que pequenos erros sejam ignorados.

A função objetivo das SVR é minimizada usando programação quadrática. Isso resulta em uma função de regressão linear no espaço de características que pode ser expressa como:

f(x) = w^T φ(x) + b

onde:

* w é um vetor de pesos
* φ(x) é a função de kernel que mapeia os dados de entrada x para o espaço de características
* b é o termo de viés

**Funções de Kernel:**

As funções de kernel comuns usadas em SVR incluem:

* **Kernel Linear:** φ(x) = x
* **Kernel Polinomial:** φ(x) = (x^T x + 1)^d
* **Kernel Gaussiano Radial:** φ(x) = e^(-γ||x - x\_i||^2)

**Parâmetros:**

As SVR têm os seguintes parâmetros:

* **C:** parâmetro de regularização que controla o grau de penalização no erro de perda
* **ε:** valor de insensibilidade que especifica a margem de erro permitida
* **γ:** parâmetro do kernel gaussiano que controla a largura da curva gaussiana

**Vantagens das SVR:**

* Robustas a outliers
* Boa generalização
* Podem lidar com dados de alta dimensão

**Desvantagens das SVR:**

* Treinamento lento para grandes conjuntos de dados
* Seleção dos parâmetros pode ser desafiadora
* Não fornece estimativas probabilísticas

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

**Ajuste de Modelos Dentro e Fora de Amostra**

* **Ajuste dentro da amostra:** O modelo é treinado usando o conjunto de dados completo disponível.
* **Ajuste fora da amostra:** O modelo é treinado usando apenas uma parte dos dados (conjunto de treinamento), enquanto a outra parte (conjunto de teste) é usada para avaliar seu desempenho.

**Overfitting**

Overfitting ocorre quando um modelo se encaixa muito bem aos dados de treinamento, mas não generaliza bem para novos dados.

* **Causas:**
  + Dados de treinamento muito pequenos ou ruidosos
  + Modelo muito complexo
* **Efeitos:**
  + Desempenho ruim em dados de teste
  + Dificuldade em interpretar ou confiar nas previsões do modelo

**Métricas de Avaliação de Overfitting**

* **Erro de treinamento:** Erro do modelo no conjunto de treinamento
* **Erro de teste:** Erro do modelo no conjunto de teste
* **Curvas de aprendizado:** Gráficos que mostram a variação do erro de treinamento e teste com o tamanho do conjunto de treinamento

**Técnicas para Evitar Overfitting**

* **Validação Cruzada:** Dividir o conjunto de dados em vários subconjuntos e treinar e avaliar o modelo em diferentes combinações desses subconjuntos.
* **Regularização:** Adicionar uma penalidade ao erro do modelo para desencorajar soluções complexas.
* **Seleção de Modelo:** Usar critérios estatísticos (por exemplo, AIC, BIC) para selecionar o modelo mais simples que se ajusta adequadamente aos dados.

Item do edital: Técnicas de agrupamento:

**Técnicas de Agrupamento**

As técnicas de agrupamento são algoritmos estatísticos que dividem um conjunto de dados em grupos (clusters) com base em suas semelhanças. Elas visam identificar padrões ou estruturas subjacentes nos dados, facilitando a compreensão e análise.

**Tipos de Técnicas de Agrupamento:**

**Técnicas Hierárquicas:**

* **Agrupamento Aglomerativo:** Começa com cada ponto como um cluster separado e gradualmente os mescla com base em sua proximidade.
* **Agrupamento Divisivo:** Começa com todos os pontos em um único cluster e divide-os gradualmente com base em sua dissimilitude.

**Técnicas Não Hierárquicas:**

* **K-Means:** Atribui pontos a k clusters iniciais e iterativamente atualiza as médias do cluster e as atribuições de pontos para minimizar a soma dos quadrados das distâncias entre os pontos e seus respectivos centróides.
* **Agrupamento de Ligação Única:** Agrupa pontos com a distância de ligação mais curta entre eles.
* **Agrupamento de Ligação Completa:** Agrupa pontos com a distância de ligação mais longa entre eles.

**Distâncias e Medidas de Semelhança:**

A escolha da distância ou medida de semelhança afeta os resultados do agrupamento. Exemplos comuns incluem:

* **Distância Euclidiana:** Distância linear entre dois pontos no espaço multidimensional.
* **Distância de Manhattan:** Soma das diferenças absolutas entre as coordenadas dos pontos.
* **Coeficiente de Correlação:** Mede a força e direção da relação linear entre duas variáveis.

**Fórmulas:**

**Distância Euclidiana:**

d(x, y) = √(Σ(x\_i - y\_i)^2)

Onde:

* x e y são dois pontos no espaço multidimensional
* x\_i e y\_i são as coordenadas dos pontos na dimensão i

**Coeficiente de Correlação de Pearson:**

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

Onde:

* x e y são dois vetores de dados
* x̄ e ȳ são as médias de x e y, respectivamente

Item do edital: Agrupamento por partição

**Agrupamento por Partição**

O agrupamento por partição, também conhecido como k-means, é um algoritmo de agrupamento não supervisionado que divide dados multidimensionais em clusters distintos.

**Como funciona:**

1. **Inicialização:** Selecione aleatoriamente k centróides (pontos representativos) para o número de clusters desejado.
2. **Atribuição:** Atribua cada ponto de dados ao cluster cuja centróide é a mais próxima.
3. **Atualização:** Calcule a nova centróide de cada cluster como a média dos pontos de dados atribuídos a ele.
4. **Repetir:** Repita as etapas 2 e 3 até que os centróides não mudem mais significativamente ou um número especificado de iterações seja alcançado.

**Fórmulas:**

* **Distância euclidiana:** calcula a distância entre dois pontos multidimensionais:

dist(p1, p2) = sqrt((p1[0] - p2[0])^2 + (p1[1] - p2[1])^2 + ... + (p1[n] - p2[n])^2)

* **Nova centróide:** calcula o centroide de um cluster como a média dos pontos atribuídos a ele:

centroid = (1/n) \* (p1 + p2 + ... + pn)

**Vantagens:**

* Simples e fácil de implementar
* Escala bem para conjuntos de dados grandes
* Pode lidar com dados multidimensionais

**Desvantagens:**

* Requer a especificação antecipada do número de clusters
* Sensível à seleção inicial de centróides
* Pode convergir para soluções locais em vez de soluções ideais

Item do edital: Agrupamento por densidade

**Agrupamento por Densidade**

O agrupamento por densidade é um método de agrupamento sem supervisão que identifica clusters com base na densidade de pontos de dados no espaço de dados.

**Conceito:**

O agrupamento por densidade identifica regiões de alta densidade (clusters) e áreas de baixa densidade (ruído). Ele atribui pontos a clusters com base em sua proximidade e distância de outros pontos.

**Fórmulas:**

* **Densidade do Ponto (ρ):** Número de pontos dentro de um raio ε do ponto dado.
* **Densidade do Cluster (Δ):** Densidade média dos pontos dentro de um cluster.
* **Acessibilidade do Ponto (σ):** Menor distância (não incluindo ε) entre um ponto e um ponto de maior densidade.

**Algoritmos:**

* **DBSCAN (Clustering Espacial com Aplicação de Ruído):** Um popular algoritmo baseado em densidade que considera os seguintes parâmetros:
  + ε: Raio de vizinhança
  + minPts: Número mínimo de pontos em uma vizinhança para considerá-la densa
* **OPTICS (Hierarquia de Ordenação de Pontos Ordenados para Clustering)**: Uma extensão do DBSCAN que cria uma árvore de alcance para identificar clusters de diferentes densidades.
* **HDBSCAN (Agrupamento por Densidade Hierárquica para Aplicações Escassas):** Uma variante do DBSCAN que lida com dados esparsos e identifica hierarquias de clusters.

**Características:**

* Identifica clusters de forma arbitrária
* Pode lidar com dados de ruído
* Não requer o número de clusters a serem especificados com antecedência
* Pode descobrir clusters hierárquicos

**Aplicações:**

* Detecção de anomalias
* Segmentação de imagens
* Análise de redes sociais
* Mineração de dados espacial

Item do edital: Agrupamento hierárquico.

**Agrupamento Hierárquico**

O agrupamento hierárquico é um método de agrupamento que cria uma representação hierárquica dos dados, construindo sequencialmente uma árvore (dendrograma) a partir de observações individuais. Cada nó da árvore representa um cluster, e o nível do nó na árvore indica a distância entre os clusters.

**Como Funciona:**

1. **Inicialização:** Cada observação é tratada como um cluster individual.
2. **Medição da Distância:** As distâncias entre os clusters são calculadas usando uma métrica de similaridade ou distância.
3. **Ligação:** Os dois clusters mais próximos (com base na distância entre eles) são ligados para formar um novo cluster.
4. **Atualização da Medição de Distância:** A distância entre o novo cluster e os clusters restantes é recalculada.
5. **Iterações:** O processo é repetido até que todos os itens sejam agrupados em um único cluster.

**Tipos de Ligação:**

* **Ligação única:** A distância entre dois clusters é determinada pela distância mais próxima entre quaisquer membros individuais dos clusters.
* **Ligação completa:** A distância entre dois clusters é determinada pela distância mais distante entre quaisquer membros individuais dos clusters.
* **Ligação média:** A distância entre dois clusters é determinada pela distância média entre todos os pares possíveis de membros entre os clusters.
* **Ligação por média ponderada:** Semelhante à ligação média, mas os pares de membros são ponderados pelo número de membros em cada cluster.
* **Ligação de Ward:** A distância entre dois clusters é determinada pela soma dos quadrados da diferença entre os membros dos clusters.

**Características:**

* Produz uma representação visual da hierarquia de dados (dendrograma).
* É fácil de implementar e interpretar.
* Pode identificar clusters de diferentes tamanhos e formas.

**Aplicações:**

* Bioinformática
* Marketing
* Análise exploratória de dados
* Classificação não supervisionada

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

**Seleção de Característica: Técnicas de Redução de Dimensionalidade**

A seleção de recursos é uma importante estratégia de redução de dimensionalidade que envolve a identificação e escolha de um conjunto reduzido de recursos informativos para treinamento de modelo.

**Objetivos:**

* Reduzir a complexidade do modelo
* Melhorar o rendimento do modelo
* Remover recursos irrelevantes ou ruidosos

**Métodos:**

Existem vários métodos de seleção de recursos, cada um com suas vantagens e desvantagens:

* **Filtragem:** Avalia recursos individualmente com base em sua relevãncia ou correlação com a variável de resposta, por ex., teste qui-quadrado, informação mútua.
* **Embutimento:** Seleciona recursos dentro do algoritmo de aprendizado, por ex., LASSO (Least absolute Shrinkage and Selection Operation), Elastic-net.
* **Envoltório:** Utiliza um algoritmo de otimização para selecionar o conjunto ideal de recursos, por ex., busca sequencial, método de Monte-Carlo Markov chain.

**Fórmulas:**

* **Teste qui-quadrado:**

χ² = Σ (Oij - Eij)^2 / Eij

onde:

* Oij é o número de observações observadas na célula ij
* Eij é o número de observações esperadas na célula ij
* **Informações mútuas:**

I(X; Y) = H(X) + H(Y) - H(X, Y)

onde:

* H(X) e H(Y) são as entropias de X e Y,
* H(X, Y) é a entropia conjunta de X e Y

**Benefícios:**

* Reduz o ru Besardo e a sobreajuste
* Aumenta a interpretabilidade do modelo
* Melhora a eficácia de treinamento

**Limitações:**

* Pode remover recursos importantes se selecionados incorretamente
* Pode ser intensivo em termos de tempo para grandes conjuntos de recursos

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

**Técnicas de Redução de Dimensionalidade: Análise de Componentes Principais (PCA)**

**Objetivo:**

Reduzir a dimensionalidade de um conjunto de dados enquanto preserva a variância essencial.

**Princípio:**

* Projeta os dados em um novo espaço ortogonal onde as primeiras direções (componentes principais) capturam a maior quantidade de variância.
* As componentes principais são combinações lineares das variáveis originais.

**Fórmulas:**

* **Covariância:**
* Cov(X, Y) = 1/n \* Σ(x - μx)(y - μy)
* **Matriz de Covariância:**
* Σ = [Cov(x\_i, x\_j)]
* **Autovalores e Autovetores:**
* det(Σ - λI) = 0
* Σv = λv

onde:

* X, Y são variáveis
* n é o número de observações
* μx, μy são as médias de X e Y
* Σ é a matriz de covariância
* λ são os autovalores
* v são os autovetores

**Etapas:**

1. Calcule a matriz de covariância.
2. Calcule os autovalores e autovetores da matriz de covariância.
3. Ordene os autovalores por magnitude decrescente e selecione os k maiores autovetores correspondentes.
4. Projete os dados originais nos k componentes principais usando os autovetores selecionados.

**Vantagens:**

* Preserva a variância máxima
* Computacionalmente eficiente
* Não requer informações de classe
* Pode ser facilmente interpretada

**Desvantagens:**

* Pode não capturar algumas informações importantes
* Não garante que a estrutura dos dados seja linear

Item do edital: Processamento de linguagem natural: Normalização textual

**Processamento de Linguagem Natural: Normalização Textual**

A normalização textual é uma etapa essencial no processamento de linguagem natural (PNL) que visa preparar o texto para análise e processamento posteriores. Envolve várias técnicas para tornar o texto mais uniforme e consistente, facilitando o processamento e melhorando a precisão dos modelos de PNL.

**Técnicas de Normalização Textual**

* **Remoção de Pontuação:** Remove pontuação desnecessária, como vírgulas, pontos e parênteses, que podem interromper o processamento.
* **Tokenização:** Divide o texto em unidades discretas chamadas tokens, geralmente palavras ou caracteres.
* **Stemming:** Reduz as palavras a sua forma raiz para melhorar a generalização e reduzir o ruído.
* **Lematização:** Reduz as palavras a um lema canônico (uma forma representativa e dicionarizada) para eliminar variações morfológicas.

**Fórmulas**

* **Fórmula de Stemming de Porter:** Um algoritmo comum de stemming que remove sufixos comuns de palavras em inglês.
* **Algoritmo de Lematização de WordNet:** Um algoritmo que usa o WordNet (um dicionário semântico) para identificar o lema de uma palavra.

**Benefícios da Normalização Textual**

* **Melhora a precisão:** Reduz ruído e variações no texto, resultando em modelos de PNL mais precisos.
* **Facilita o processamento:** Cria um texto mais uniforme, permitindo que os algoritmos de PNL processem o texto de forma mais eficiente.
* **Aumenta a generalização:** Permite que os modelos de PNL generalizem para novos textos, mesmo que contenham variações linguísticas.
* **Reduz a complexidade:** Torna o texto mais gerenciável e reduz a complexidade computacional durante o processamento.

**Conclusão**

A normalização textual é uma etapa crucial na PNL, pois prepara o texto para processamento posterior e melhora a precisão dos modelos de PNL. As técnicas de normalização, como tokenização, stemming e lematização, transformam o texto em uma forma mais uniforme e consistente, facilitando a análise e garantindo modelos de alta qualidade.

Item do edital: Processamento de linguagem natural: stop words

**Processamento de Linguagem Natural: Stop Words**

As palavras de parada (stop words) são palavras comuns que ocorrem com alta frequência em um idioma, mas que geralmente não transmitem muito significado ou valor informativo específico. Elas são removidas do texto durante o processamento de linguagem natural (PNL) para melhorar a eficiência dos algoritmos de PNL.

**Razões para Remover Stop Words:**

* **Redução da dimensionalidade:** As stop words representam uma grande proporção do texto, mas não contribuem significativamente para o significado. Sua remoção reduz a dimensionalidade do texto.
* **Melhoria da precisão:** As stop words podem causar ruído nos algoritmos de PNL, tornando mais difícil extrair recursos informativos do texto.
* **Aceleração do processamento:** Remover stop words reduz o tamanho do texto, acelerando os algoritmos de PNL.

**Fórmulas:**

A frequência de uma palavra de parada (f) pode ser calculada usando a fórmula:

f = N(w) / N

onde:

* N(w) é o número de ocorrências da palavra w no texto
* N é o número total de palavras no texto

**Exemplos Comuns de Stop Words:**

Em inglês, alguns exemplos comuns de stop words incluem:

* a, an, the
* of, to, in
* is, are, was

**Métodos de Remoção de Stop Words:**

* **Listas de stop words:** Listas pré-compiladas de stop words são usadas para identificar e remover essas palavras do texto.
* **Remoção estatística:** Palavras com frequência extremamente alta (por exemplo, f > 0,9) são consideradas stop words e removidas.
* **Aprendizagem de máquina:** Algoritmos de aprendizado de máquina podem ser treinados para identificar stop words com base em dados de texto.

Item do edital: Processamento de linguagem natural: estemização

**Processamento de Linguagem Natural: Estemização**

A estemização é um processo de redução de palavras a sua forma raiz ou radical, conhecida como esteme. Este processo remove sufixos e prefixos não essenciais, simplificando a palavra para melhor análise e processamento.

A estemização é normalmente realizada usando algoritmos que seguem regras linguísticas pré-definidas. Alguns dos algoritmos mais comuns incluem:

* **Algoritmo de Porter:** Um algoritmo popular que remove sufixos comuns e aplica regras de exceção para produzir esteme.
* **Algoritmo Lovins:** Semelhante ao algoritmo de Porter, mas mais abrangente e mais adequado para textos técnicos.
* **Algoritmo de Lancaster:** Um algoritmo mais complexo que manipula palavras usando uma estrutura de árvore.

**Objetivos da Estemização:**

* Normalizar palavras para melhorar a recuperação e agrupamento de documentos.
* Reduzir a complexidade do vocabulário, tornando o processamento mais eficiente.
* Identificar conceitos ou ideias subjacentes independentemente da variação de palavras.

**Exemplo:**

Considerando a palavra "dançando", a sua estemização resultaria em "danç".

**Fórmulas:**

As fórmulas específicas usadas nos algoritmos de estemização são complexas e envolvem regras linguísticas e exceções. No entanto, uma fórmula geral para remover um sufixo pode ser representada como:

palavra[0:n]

Onde:

* **palavra[0:n]** é a subcadeia da palavra do início (0) até antes da posição (n) do sufixo a ser removido.

Item do edital: Processamento de linguagem natural: lematização

**Processamento de Linguagem Natural: Lematização**

A lematização é uma técnica de Processamento de Linguagem Natural (PLN) que converte palavras flexionadas ou derivados de palavras na sua forma canónica ou lema. Ao remover sufixos e prefixos, a lematização ajuda a normalizar o texto para facilitar o processamento e a análise.

**Objetivos da Lematização:**

* Reduzir a redundância lexical
* Melhorar a correspondência de padrões
* Aumentar a precisão das tarefas de PLN

**Fórmulas Comuns de Lematização:**

* **Algoritmo de Porter:** Um algoritmo amplamente utilizado que remove sufixos comuns à medida que percorre a palavra do fim para o início.
* **Algoritmo de Lancaster:** Uma variante do algoritmo de Porter que lida melhor com palavras derivadas de verbos.
* **WordNet Lemmatizer:** Uma ferramenta baseada em dicionário que usa o WordNet para identificar a forma canónica das palavras.

**Como Funciona a Lematização:**

A lematização envolve os seguintes passos:

1. **Tokenização:** Dividir o texto em palavras individuais.
2. **Remoção de Sufixos e Prefixos:** Identificar e remover sufixos e prefixos comuns que não alterem o significado da palavra.
3. **Identificação do Lema:** Consultar um dicionário ou algoritmo para encontrar a forma canónica da palavra.

**Aplicações da Lematização:**

A lematização é útil em várias tarefas de PLN, incluindo:

* Indexação e recuperação de informação
* Análise de sentimentos
* Resumo de texto
* Reconhecimento de entidades nomeadas

**Conclusão:**

A lematização é uma técnica essencial de PLN que normaliza o texto ao converter palavras flexionadas para as suas formas canónicas. Isso melhora a precisão das tarefas de PLN ao reduzir a redundância e facilitar o processamento.

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

**Análise de Frequência de Termos em Processamento de Linguagem Natural**

A análise de frequência de termos é uma técnica fundamental em Processamento de Linguagem Natural (PNL) que visa analisar a ocorrência de termos específicos em um conjunto de texto. Ela quantifica a importância relativa dos termos no texto, fornecendo insights sobre o conteúdo e a estrutura do texto.

**Procedimento:**

1. **Tokenização:** Dividir o texto em unidades individuais (tokens), como palavras ou n-gramas.
2. **Remoção de stop words:** Remover palavras comuns irrelevantes, como "o", "a" e "para".
3. **Stemming ou Lematização:** Reduzir as palavras a suas formas básicas (ou seja, raiz ou lema).
4. **Contagem de frequência:** Contar o número de ocorrências de cada termo exclusivo.

**Fórmulas:**

**Frequência Relativa (RF):**

RF(t) = (N(t) / N) \* 100

onde:

* RF(t) é a frequência relativa do termo t
* N(t) é o número de ocorrências de t
* N é o número total de tokens no texto

**Frequência Inversa de Documentos (IDF):**

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

onde:

* IDF(t) é a frequência inversa de documentos do termo t
* N é o número total de documentos no corpus
* df(t) é o número de documentos que contêm o termo t

**TF-IDF:**

O escore TF-IDF é o produto da frequência relativa e da frequência inversa de documentos, que combina a importância do termo no documento individual e sua singularidade no corpus.

TF-IDF(t) = RF(t) \* IDF(t)

**Aplicações:**

* Classificação de texto
* Extração de palavras-chave
* Resumo de texto
* Detecção de plágio

**Benefícios:**

* Fornece uma representação numérica do conteúdo do texto
* Identifica termos importantes e tópicos emergentes
* Ajuda a entender a estrutura e a organização do texto

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

**Rotulação de Partes do Discurso**

A rotulação de partes do discurso (POS) é uma tarefa de processamento de linguagem natural que atribui uma etiqueta de categoria gramatical (parte do discurso) a cada palavra em uma frase.

**Objetivos**

* Fornecer informações linguísticas estruturais
* Melhorar a precisão de tarefas posteriores de PNL, como análise sintática e semântica

**Técnicas**

* **Baseadas em regras:** Usam conjuntos pré-definidos de regras para atribuir etiquetas.
* **Baseadas em estatísticas:** Usam modelos estatísticos treinados em dados rotulados para prever etiquetas.
* **Modelos híbridos:** Combinam técnicas baseadas em regras e estatísticas.

**Etiquetas Comuns**

* Substantivo (N)
* Verbo (V)
* Adjetivo (A)
* Advérbio (ADV)
* Preposição (P)
* Determinante (DET)
* Pronome (PRON)
* Conjunção (CONJ)
* Interjeição (INT)

**Medidas de Avaliação**

* Precisão: Porcentagem de etiquetas atribuídas corretamente
* Revocação: Porcentagem de etiquetas corretas identificadas

**Fórmulas**

* **Precisão:** Precisão = Etiquetas corretas / Etiquetas atribuídas
* **Revocação:** Revocação = Etiquetas corretas / Etiquetas esperadas

**Aplicações**

* Análise sintática
* Reconhecimento de entidade nomeada
* Resumo de texto
* Tradução automática

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

**Modelos de Representação de Texto: N-gramas**

Os **N-gramas** são um modelo de representação de texto que fragmenta o texto em subsequências de comprimento fixo. Cada subsequência é chamada de n-grama.

**Fórmula:**

Um n-grama é uma sequência de **n** tokens consecutivos:

n-grama = (token₁, token₂, ..., tokenₙ)

**Tipos de N-gramas:**

* **Unigramas (n=1):** Tokens individuais.
* **Bigramas (n=2):** Pares de tokens.
* **Trigramas (n=3):** Tercetos de tokens.
* **N-gramas de ordem superior:** Sequências mais longas de tokens.

**Características:**

* Capturam sequências ordenadas de tokens.
* Podem ser usados para modelagem de linguagem, classificação de texto e tradução automática.
* Os n-gramas de ordem superior captam relacionamentos mais complexos entre tokens.
* Podem ser representados como vetores ou matrizes.
* São sensíveis à ordem dos tokens, o que pode ser uma vantagem ou desvantagem dependendo da tarefa.

**Vantagens:**

* Simples e fácil de implementar.
* Podem identificar padrões locais no texto.

**Desvantagens:**

* Podem ser esparsos, especialmente para n-gramas de ordem superior.
* Podem ser afetados por dados esparsos, o que pode levar a problemas de superajuste.
* Não capturam a estrutura hierárquica do texto.

**Conclusão:**

Os N-gramas são um modelo de representação de texto amplamente utilizado que pode capturar relacionamentos ordenados entre tokens. Embora seus recursos sejam limitados em comparação com modelos mais avançados, eles permanecem uma ferramenta valiosa para várias tarefas de processamento de linguagem natural.

Item do edital: modelos vetoriais de palavras: CBOW

**Modelos Vetoriais de Palavras: CBOW (Continuous Bag-of-Words)**

**Introdução:** O CBOW é um modelo vetorial de palavras que prevê a palavra atual (palavra-alvo) com base nas palavras de contexto (palavras próximas). Ele aprende vetores de palavras contínuos, representando o significado semântico das palavras.

**Arquitetura:** O CBOW tem uma arquitetura de entrada-saída com as seguintes camadas:

* **Camada de Entrada:** Recebe uma janela de palavras de contexto (por exemplo, 5 palavras à esquerda e à direita).
* **Camada de Projeção:** Projeta as palavras de contexto em embeddings vetoriais.
* **Camada de Soma:** Soma os embeddings das palavras de contexto.
* **Camada de Saída:** Prediz a palavra-alvo usando uma camada densa (por exemplo, softmax).

**Fórmulas:**

* **Embedding da Palavra de Contexto:** (e\_c = W\_c \times c)
* **Soma dos Embeddings:** (x = \sum\_{c=1}^{m} e\_c)
* **Previsão da Palavra-Alvo:** (y = softmax(W\_o \times x + b\_o))

Onde:

* (c) é a palavra de contexto
* (m) é o número de palavras de contexto
* (W\_c) é a matriz de projeção
* (x) é o vector de soma dos embeddings
* (W\_o) e (b\_o) são os pesos e bias da camada de saída

**Benefícios:**

* **Embeddings Semânticos:** Captura o significado semântico das palavras, permitindo comparações e agrupamentos baseados na similaridade.
* **Previsões Rápidas:** O modelo é eficiente para treinar e usar, mesmo com grandes conjuntos de dados.
* **Captura o Contexto:** Considere as palavras de contexto para prever a palavra-alvo, fornecendo representações mais ricas das palavras.

**Desvantagens:**

* **Sequência Ignorada:** Não considera a ordem das palavras de contexto.
* **Palavras Raras:** Pode ter dificuldades para representar palavras raras que não aparecem com frequência suficiente no corpus de treinamento.

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

**Modelos Vetoriais de Palavra: Skip-Gram**

O Skip-Gram é um modelo vetorial de palavra que captura relacionamentos semânticos e contextuais entre palavras em um texto. Ele foi proposto por Tomas Mikolov et al. (2013).

**Funcionamento:**

O Skip-Gram prevê a ocorrência de uma palavra (palavra-alvo) com base nas palavras que a rodeiam (palavras de contexto). Dado um conjunto de palavras de contexto {w\_1, ..., w\_k}, o modelo tenta prever a palavra-alvo w\_t de acordo com a fórmula:

P(w\_t | w\_1, ..., w\_k) = softmax(U'v\_t)

onde:

* U é uma matriz de projeção
* v\_t é o vetor embutido da palavra-alvo w\_t
* softmax() é a função softmax que normaliza as probabilidades de saída

**Treinamento:**

O modelo Skip-Gram é treinado usando retropropagação para minimizar a perda de entropia cruzada:

loss = -∑\_t log P(w\_t | w\_1, ..., w\_k)

**Benefícios:**

* Captura relacionamentos semânticos e contextuais
* Gerenciável, mesmo para conjuntos de dados grandes
* Rápido de treinar
* Fácil de implementar

**Aplicações:**

* Processamento de Linguagem Natural (PNL)
* Aprendizagem de Máquina Supervisionada
* Modelagem de Tópicos
* Resumo de Texto
* Geração de Linguagem Natural

Item do edital: modelos vetoriais de palavra: GloVe

**Modelos Vetoriais de Palavras: GloVe**

O Global Vectors for Word Representation (GloVe) é um modelo de representação de palavras que captura as relações semânticas e sintáticas das palavras. Ele foi desenvolvido no Stanford NLP Group em 2014.

**Funcionamento:**

O GloVe é treinado em um grande corpus de texto, como o Wikipedia. Ele constrói um modelo que:

* Prediz a probabilidade de ocorrência de uma palavra, dada a ocorrência de outras palavras em uma janela de contexto.
* Usa uma função de perda de mínimos quadrados ponderados (L2) para minimizar a soma das diferenças quadradas entre as probabilidades previstas e reais.

**Características:**

* **Captura semântica e sintaxe:** O GloVe considera tanto coocorrências globais (em todo o corpus) quanto locais (em janelas de contexto), permitindo que ele capture relações semânticas e sintáticas.
* **Codificação de dimensão baixa:** Os vetores GloVe são tipicamente codificados em uma dimensão baixa (e.g., 100, 300), facilitando seu uso em aplicações de aprendizado de máquina.
* **Treinamento eficiente:** O GloVe é treinado usando algoritmos eficientes, como a descida do gradiente estocástico (SGD).

**Aplicações:**

Os vetores GloVe são amplamente utilizados em uma variedade de aplicações de processamento de linguagem natural, incluindo:

* Classificação de texto
* Gerador de linguagem natural
* Resumo de texto

**Fontes:**

* Jeffrey Pennington, Richard Socher, Christopher Manning. "GloVe: Global Vectors for Word Representation." EMNLP 2014.

Item do edital: modelos vetoriais de documentos: booleano

**Modelos Vetoriais Booleanos de Documentos**

Os modelos vetoriais booleanos representam documentos como vetores binários, onde cada componente representa a presença ou ausência de um termo específico.

**Fórmula:**

d = (e1, e2, ..., en)

Onde:

* d é o vetor binário do documento
* ei é 1 se o termo i está presente no documento, senão 0

**Características:**

* Cada documento é representado como um ponto no espaço vetorial booleano.
* A similaridade entre documentos é medida pela **medida de similaridade de Jaccard**, que é calculada como a proporção de termos comuns:

similaridade(d1, d2) = |d1 ∩ d2| / |d1 ∪ d2|

**Vantagens:**

* Simples e eficiente de calcular.
* Fácil de interpretar, pois cada componente representa um termo específico.

**Desvantagens:**

* Não considera a frequência dos termos ou a ordem das palavras.
* Pode ser esparso, levando a uma baixa similaridade entre documentos com muitos termos em comum.

**Aplicações:**

* Pesquisa de documentos binários (por exemplo, documentos de metadados)
* Sistemas de recuperação de informações onde a precisão é crucial

Item do edital: modelos vetoriais de documentos: TF

**Modelos Vetoriais de Documentos: TF (Frequência de Termos)**

Os modelos vetoriais de documentos representam documentos como vetores em um espaço n-dimensional, onde n é o número de termos únicos no corpus. TF (Frequência de Termos) é uma medida do número de ocorrências de um determinado termo em um documento.

**Fórmula:**

TF(t, d) = número de ocorrências do termo t no documento d / número total de termos no documento d

**Propriedades:**

* Mede a importância relativa de um termo dentro de um documento específico.
* Não considera a importância global do termo no corpus.
* Valores mais altos indicam maior relevância do termo para o documento.
* Pode ser modificado para dar maior peso a termos que aparecem no início ou final do documento.

**Limitações:**

* Não captura a ordem dos termos no documento.
* Não considera o contexto dos termos.
* Pode ser distorcido por documentos muito longos ou curtos.

**Aplicações:**

* Indexação de documentos
* Recuperação de informações
* Agrupamento de documentos

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

**Modelos Vetoriais de Documentos: TF-IDF**

Os Modelos Vetoriais de Documentos (TF-IDF) são representações matemáticas de documentos que capturam sua similaridade usando vetores numéricos. Eles são amplamente utilizados em recuperação de informações e análise de texto.

**TF - Frequência de Termo**

A frequência de termo (TF) mede o número de ocorrências de um termo em um documento. É calculado como:

TF(t, d) = número de ocorrências do termo t no documento d

**IDF - Frequência Inversa do Documento**

A frequência inversa do documento (IDF) mede o quão comum um termo é na coleção de documentos. Ele é calculado como:

IDF(t, D) = log((|D| + 1) / (df(t) + 1))

onde:

* |D| é o número de documentos na coleção
* df(t) é o número de documentos que contêm o termo t

**Peso TF-IDF**

O peso TF-IDF é o produto da TF e IDF. Ele mede a importância de um termo em um documento específico em relação à coleção inteira. É calculado como:

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

**Vantagens:**

* Captura a importância relativa dos termos em um documento
* Reduz a distorção causada por termos comuns
* Permite a comparação e busca de documentos

**Limitações:**

* Não considera a ordem ou proximidade dos termos
* Pode ser sensível a termos raros ou muito comuns

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

**Modelos Vetoriais de Documentos: Média de Vetores de Palavras**

Os modelos vetoriais são uma representação numérica de documentos que capturam semelhanças semânticas. O modelo de vetores de palavras médias é um tipo de modelo vetorial que representa um documento como um vetor, onde cada elemento é a média dos vetores de palavras que ocorrem no documento.

Seja d um documento, e v\_i o vetor de palavras para a palavra i no vocabulário V. O modelo de vetores de palavras médias m\_d para d é definido como:

m\_d = (1/|d|) \* Σ(v\_i, i ∈ d)

onde |d| é o número de palavras em d.

**Vantagens:**

* Captura relações semânticas entre palavras
* Simples de calcular e implementar
* Efetivo para tarefas de classificação de documentos

**Desvantagens:**

* Ignora a ordem das palavras e a sintaxe
* Pode ser sensível a palavras raras e comuns

Item do edital: modelos vetoriais de documentos: Paragraph Vector

**Modelos Vetoriais de Documentos: Paragraph Vector**

Os modelos vetoriais de documentos são usados para representar documentos como vetores em um espaço vetorial de alta dimensão. Um exemplo notável é o Paragraph Vector, proposto por Le e Mikolov em 2014.

**Princípio Fundamental:**

O Paragraph Vector assume que os parágrafos em um documento estão próximos no espaço semântico. Ele aprende representações vetoriais fixas para parágrafos usando uma rede neural convolucional.

**Arquitetura da Rede:**

A arquitetura típica do Paragraph Vector consiste nas seguintes camadas:

* **Camada de Embutimento:** Embuta cada palavra em um vetor.
* **Camada Convolucional:** Aplica filtros convolucionais sobre os vetores de embutimento para extrair recursos.
* **Camada de Max Pooling:** Seleciona o recurso máximo de cada janela convolucional.
* **Camada Linear:** Projeta os vetores max pooling em um espaço vetorial de dimensão fixa.

**Fórmulas:**

A saída da camada linear (vetor de representação do parágrafo) é calculada como:

p = W \* max(conv(x)) + b

onde:

* p é o vetor de representação do parágrafo
* W é a matriz de projeção
* b é o vetor de deslocamento
* conv(x) é a saída da camada convolucional
* max() é a operação de max pooling

**Aplicações:**

Os Paragraph Vectors podem ser usados em várias aplicações de Processamento de Linguagem Natural, incluindo:

* Recuperação de informações
* Classificação de documentos
* Resumo automático
* Modelagem de tópicos

**Vantagens:**

* Captura relações semânticas entre parágrafos.
* Pode ser usado para representar documentos longos com estruturas complexas.
* Produz representações fixas e de comprimento fixo.

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

**Métricas de Similaridade Textual: Similaridade do Cosseno**

A similaridade do cosseno é uma medida de similaridade entre dois vetores que representam documentos textuais. É calculada como o cosseno do ângulo entre os vetores:

Sim Cos = Cos(θ) = (a · b) / (||a|| ||b||)

onde:

* a e b são os vetores de representação do documento
* · é o produto escalar
* ||a|| e ||b|| são as normas euclidianas dos vetores

**Como funciona:**

A similaridade do cosseno mede a extensão pela qual os vetores apontam na mesma direção. Um valor alto indica que os vetores são semelhantes, enquanto um valor próximo de zero indica que eles são dissimilares.

Os vetores de representação do documento são geralmente gerados usando métodos como o bag-of-words (BOW) ou o modelo de tópicos latentes (LSA). Esses métodos atribuem pesos às palavras no texto, criando vetores que representam a importância relativa das palavras em cada documento.

**Vantagens:**

* **Interpretabilidade:** A similaridade do cosseno é intuitiva e fácil de interpretar. Um valor alto indica similaridade, enquanto um valor baixo indica dissimilaridade.
* **Robustez:** A similaridade do cosseno é relativamente robusta a diferenças na frequência das palavras, tornando-a adequada para textos com domínios de vocabulário variados.
* **Baixo custo computacional:** A similaridade do cosseno é eficiente para calcular, o que a torna adequada para aplicações em larga escala.

**Desvantagens:**

* **Não considera a ordem das palavras:** O BOW não considera a ordem das palavras, o que pode levar a vetores de representação semelhantes para documentos com significados diferentes.
* **Pode ser sensível ao comprimento do texto:** A similaridade do cosseno pode ser tendenciosa em relação a documentos mais longos.
* **Pode produzir resultados inesperados:** Em alguns casos, textos muito diferentes podem ter uma similaridade do cosseno alta devido à distribuição de palavras com pesos semelhantes.

Item do edital: Métricas de similaridade textual distância euclidiana

**Métricas de Similaridade Textual**

As métricas de similaridade textual medem a semelhança entre dois textos, levando em consideração fatores como frequência de palavras, ordem das palavras e estrutura sintática. Elas são usadas em várias aplicações, como agrupamento de documentos, recuperação de informações e tradução automática.

**Distância Euclidiana**

A distância euclidiana entre dois vetores **x** e **y** é definida como:

d(x, y) = √(Σ(xᵢ - yᵢ)²)

onde:

* **x** e **y** são vetores com o mesmo número de dimensões
* **xᵢ** e **yᵢ** são os valores do **i**-ésimo elemento em **x** e **y**, respectivamente

**Aplicação à Similaridade Textual**

Para medir a similaridade textual usando a distância euclidiana, os textos são representados como vetores de frequência de palavras. Os vetores são então subtraídos um do outro e a distância entre eles é calculada usando a fórmula acima.

Quanto menor a distância, mais semelhantes são os textos. Isso ocorre porque a distância euclidiana mede a diferença entre as frequências das palavras nos dois textos.

**Vantagens e Desvantagens**

* **Vantagens:**
  + Simples de calcular
  + Pode detectar diferenças significativas na frequência das palavras
* **Desvantagens:**
  + Ignora a ordem das palavras e a estrutura sintática
  + Associa uma alta penalidade a diferenças nas frequências das palavras

**Aplicações**

A distância euclidiana é comumente usada em:

* Filtragem de spam de e-mail
* Grupo de documentos
* Detecção de plágio

Item do edital: Métricas de similaridade textual similaridade de Jaccard

**Métricas de Similaridade Textual: Similaridade de Jaccard**

A similaridade de Jaccard é uma métrica amplamente utilizada para medir a similaridade entre dois conjuntos de texto. Ela calcula a proporção de elementos comuns (palavras ou termos) em relação ao número total de elementos distintos entre os dois conjuntos.

**Fórmula:**

Similaridade de Jaccard = |A ⋂ B| / |A ∪ B|

onde:

* |A ⋂ B| é o número de elementos comuns aos dois conjuntos
* |A ∪ B| é o número total de elementos distintos nos dois conjuntos

**Interpretação:**

Uma pontuação de similaridade de Jaccard de 0 indica que os conjuntos não são semelhantes, enquanto um valor de 1 indica que os conjuntos são idênticos. Pontuações entre 0 e 1 medem diferentes níveis de similaridade.

**Aplicações:**

A similaridade de Jaccard é usada em vários aplicativos de processamento de linguagem natural, incluindo:

* **Busca de informações:** Para recuperar documentos semelhantes a uma consulta
* **Agrupamento de texto:** Para agrupar documentos com base em sua similaridade textual
* **Detecção de plágio:** Para identificar textos que são substancialmente semelhantes

**Vantagens e Desvantagens:**

* **Vantagens:**
  + Simples de calcular
  + Insensível à ordem dos elementos
* **Desvantagens:**
  + Pode ser sensível ao tamanho do conjunto
  + Não considera a similaridade semântica entre palavras ou termos

Item do edital: Métricas de similaridade textual distância de Manhattan

:.

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

**Coeficiente de Dice**

O coeficiente de Dice é uma métrica de similaridade textual usada para medir o grau de sobreposição entre dois conjuntos de tokens (palavras ou elementos distintos). É definido como:

Coeficiente de Dice = 2 \* Interseção(A, B) / (|A| + |B|)

onde:

* A e B são os conjuntos de tokens
* Interseção(A, B) é o número de tokens comuns a ambos os conjuntos
* |A| e |B| são os tamanhos dos conjuntos A e B, respectivamente

**Interpretação**

O coeficiente de Dice varia de 0 a 1. Um valor de 1 indica que os conjuntos são idênticos, enquanto um valor de 0 indica que não há sobreposição.

**Aplicações**

O coeficiente de Dice é usado em várias aplicações, incluindo:

* **Recuperação de informações:** Medir a semelhança entre documentos ou consultas
* **Processamento de linguagem natural:** Agrupamento de texto, extração de entidades e resumo
* **Ciência da computação:** Detecção de plágio, comparação de algoritmos e análise de código-fonte

**Vantagens**

* Simples de calcular
* Robust a variações na ordem dos tokens
* Maneira útil de quantificar a sobreposição de conjuntos de tokens

**Desvantagens**

* Não considera a distância entre os tokens
* Pode ser afetado pelo comprimento dos conjuntos
* Não leva em consideração a frequência dos tokens

Item do edital: Redes neurais convolucionais

**Redes Neurais Convolucionais (CNNs)**

As Redes Neurais Convolucionais são um tipo de rede neural artificial especializada no processamento de dados grid-like, como imagens e matrizes. Elas são caracterizadas por operações de convolução que capturam padrões locais de dados.

**Operações de Convolução:**

A convolução é uma operação matemática que envolve uma matriz de filtro (kernel) que se desliza sobre a entrada, computando uma soma ponderada dos valores sobrepostos:

Convolução (X, W) = ΣΣ X(i, j) \* W(k, l)

* X é a entrada (matriz)
* W é o kernel
* k e l são os índices do kernel
* i e j são os índices da entrada

**Arquitetura das CNNs:**

As CNNs são tipicamente compostas de camadas alternadas de:

* **Camadas convolucionais:** Aplicam a operação de convolução para extrair características locais.
* **Camadas de pooling:** Reduzem a dimensionalidade dos mapas de características usando funções como max-pooling ou média-pooling.
* **Camadas totalmente conectadas:** Classificam ou regridem as saídas das camadas convolucionais.

**Aplicações das CNNs:**

As CNNs são amplamente utilizadas em uma variedade de aplicações, incluindo:

* Reconhecimento de imagem e objeto
* Processamento de linguagem natural
* Detecção biomédica e análise de dados
* Previsão de séries temporais

**Benefícios das CNNs:**

* **Extração automática de características:** As CNNs podem aprender automaticamente características relevantes dos dados, eliminando a necessidade de engenharia de características.
* **Tolerância a variações:** As camadas convolucionais são invariantes a pequenas traduções e rotações dos dados de entrada.
* **Alta precisão:** As CNNs demonstraram níveis excepcionais de precisão em diversas tarefas.

**Limitações das CNNs:**

* **Requisitos de dados:** As CNNs geralmente requerem conjuntos de dados grandes para treinamento.
* **Alta complexidade computacional:** As operações de convolução podem ser computacionalmente caras.
* **Interpretabilidade:** As CNNs podem ser difíceis de interpretar, tornando-as menos transparentes do que modelos mais simples.

Item do edital: Redes neurais recorrentes.

**Redes Neurais Recorrentes (RNNs)**

As RNNs são uma classe de redes neurais artificiais projetadas para processar dados sequenciais, como texto, fala e dados de séries temporais. Ao contrário das redes neurais feedforward tradicionais, as RNNs possuem conexões recorrentes, o que permite que elas levem informações de entradas anteriores para entradas subsequentes.

**Fórmulas Básicas:**

* **Célula de Memória:** Cada célula de memória em uma RNN mantém um estado interno, representado por **h**.
* **Função de Transição:** A função de transição atualiza o estado da célula com base na entrada atual **x** e no estado anterior **h(t-1)**:
  + **h(t) = f(Wx + Uh(t-1))**

Onde: \* **W** e **U** são matrizes de pesos \* **f** é uma função não linear (por exemplo, tanh ou ReLU)

* **Saída:** A saída da RNN é computada a partir do estado atual da célula de memória:
  + **y(t) = g(Vh(t))**

Onde: \* **V** é uma matriz de pesos \* **g** é uma função de ativação (por exemplo, sigmoid ou softmax)

**Tipos de RNNs:**

* **RNNs de Tempo Descontínuo (DRNNs):** As DRNNs processam dados sequenciais com comprimentos variáveis.
* **RNNs Bidirecionais (BRNNs):** As BRNNs processam dados sequenciais em ambas as direções, fornecendo informações de contexto bidirecional.
* **LSTM (Memória de Longo Prazo):** As LSTMs são RNNs com células de memória aprimoradas que podem lidar com dependências de longo prazo.

**Aplicações:**

As RNNs são amplamente utilizadas em várias aplicações, incluindo:

* Processamento de Linguagem Natural
* Reconhecimento de Fala
* Geração de Texto
* Previsão de Séries Temporais

Item do edital: Scikit-learn

**Scikit-learn: Uma Biblioteca Abrangente para Aprendizado de Máquina em Python**

Scikit-learn é uma biblioteca de aprendizado de máquina de código aberto para Python que fornece uma ampla gama de algoritmos e ferramentas para tarefas de aprendizado de máquina. Ele é projetado para ser eficiente, fácil de usar e extensível.

**Principais Recursos:**

* **Algoritmos de Aprendizado Supervisionado:** Classificação (e.g., Árvores de Decisão, Regressão Logística), Regressão (e.g., Regressão Linear, Árvores de Regressão)
* **Algoritmos de Aprendizado Não Supervisionado:** Clustering (e.g., K-Means, Hierárquico), Redução de Dimensionalidade (e.g., PCA)
* **Pré-processamento e Transformação de Dados:** Padronização, Normalização, Codificação de Recursos
* **Avaliação de Modelos:** Métricas de Desempenho (e.g., Precisão, Recuo), Seleção de Modelos, Validação Cruzada
* **Integração com Outros Ecossistemas:** Suporta integração com bibliotecas populares como NumPy, Pandas e Matplotlib

**Destaques:**

* **Eficiência:** Algoritmos otimizados para desempenho.
* **Facilidade de Uso:** Interfaces de usuário simples e documentação abrangente.
* **Extensibilidade:** Codificação orientada a objetos permite a criação de novos algoritmos e transformações.

**Uso:**

Scikit-learn pode ser usado para uma ampla variedade de tarefas de aprendizado de máquina, incluindo:

* **Classificação:** Prever categorias (e.g., spam/legítimo, gato/cão)
* **Regressão:** Prever valores contínuos (e.g., preço da casa, temperatura)
* **Clustering:** Agrupar dados em grupos semelhantes
* **Redução de Dimensionalidade:** Reduzir a complexidade dos dados, mantendo a informação relevante

**Fórmulas:**

* **Classificação Logística:**

p = 1 / (1 + exp(-w^T x))

onde: \* p é a probabilidade de pertencer à classe 1 \* w é o vetor de pesos \* x é o vetor de características

* **Árvores de Regressão:**

y\_pred = sum(alpha\_i \* h(x, s\_i))

onde: \* y\_pred é a previsão \* alpha\_i são os pesos dos nós folhas \* h(x, s\_i) é a função de partição do nó folha com divisão s\_i

Item do edital: TensorFlow

**TensorFlow**

TensorFlow é uma plataforma de aprendizado de máquina de código aberto desenvolvida pelo Google. É usado para construir e treinar modelos de aprendizado de máquina para uma ampla gama de tarefas, incluindo:

* **Aprendizado supervisionado:** Classificação, regressão
* **Aprendizado não supervisionado:** Clustering, redução de dimensionalidade
* **Aprendizado por reforço:** Jogos, automação

**Arquitetura**

O TensorFlow é construído em torno do conceito de **tensores**, que são arrays multidimensionais. Os tensores fluem através de um **gráfico computacional**, que define as operações a serem executadas nos dados. O gráfico é otimizado e executado por um mecanismo de execução.

**Fórmulas**

* **Gradiente:** A derivada da função de perda em relação aos parâmetros do modelo.
* **Passo de gradiente:** O tamanho do passo dado na direção do gradiente descendent.
* **Função de perda:** Uma função que mede o erro do modelo.
* **Taxa de aprendizado:** Controla o tamanho dos passos de gradiente.

**Características**

* **Flexibilidade:** Suporta uma ampla gama de modelos e algoritmos.
* **Escalável:** Pode treinar modelos em grandes conjuntos de dados usando vários dispositivos.
* **Eficiente:** Otimiza o código para acelerar o treinamento e a inferência.
* **Comunidade ativa:** Grande comunidade de usuários e contribuidores.

**Códigos de exemplo**

# Cria um tensor com os valores [1, 2, 3]

import tensorflow as tf

x = tf.constant([1, 2, 3])

# Multiplica o tensor por 2

y = x \* 2

# Exibe o tensor resultante

print(y)

**Aplicações**

O TensorFlow tem sido usado em uma ampla gama de aplicações, incluindo:

* Reconhecimento de imagem
* Processamento de linguagem natural
* Previsão de séries temporais
* Aprendizado por reforço
* Bioinformática

Item do edital: PyTorch

**O que é PyTorch?**

PyTorch é uma estrutura de aprendizado profundo de código aberto, baseada em Python, desenvolvida pelo Facebook AI Research (FAIR). É especializada em computação com tensores e diferenciação automática.

**Recursos Principais:**

* **Computação com Tsores:** PyTorch permite trabalhar com tensores multidimensionais, representando dados numéricos.
* **Diferenciação Automática:** A biblioteca fornece gradientes calculados automaticamente por meio da retropropagação, facilitando o treinamento de modelos.
* **Modelos Flexíveis:** Os modelos no PyTorch são construídos usando módulos e otimizadores personalizáveis.
* **Integração de GPU:** PyTorch suporta treinamento acelerado em GPUs para melhorar o desempenho computacional.
* **Ampla Comunidade:** Possui uma comunidade ativa e ampla documentação, tornando-o um recurso valioso para pesquisadores e praticantes de aprendizado profundo.

**Fórmulas:**

* **Propagação:** y = f(x)
* **Retropropagação:** dL/dx = (dL/dy) \* (dy/dx)
* **Otimização por Gradiente Descendente:** w = w - lr \* (dL/dw) (onde w é o peso, lr é a taxa de aprendizado)

**Aplicações:**

* Visão computacional (detecção de objetos, classificação de imagens)
* Processamento de linguagem natural (tradução de máquina, reconhecimento de fala)
* Aprendizado de reforço (robótica, jogos)
* Finanças quantitativas (previsão de séries temporais, risco)

**Conclusão:**

PyTorch é uma estrutura poderosa e versátil para aprendizado profundo, oferecendo computação de tensor eficiente, diferenciação automática e modelos flexíveis. É amplamente utilizado em vários domínios para construir e treinar modelos de aprendizado de máquina de última geração.

Item do edital: Keras

**O que é Keras?**

Keras é uma biblioteca de rede neural de alto nível construída sobre TensorFlow. Ela foi projetada para ser fácil de usar, modular e extensível.

**Características:**

* **Interface intuitiva:** Keras usa uma API concisa e fácil de entender.
* **Construção modular:** As redes neurais podem ser construídas unindo camadas individuais, oferecendo flexibilidade.
* **Suporte a várias GPUs:** Keras permite o treinamento em paralelo em GPUs para acelerar o processo.
* **Ampla gama de camadas:** Keras oferece uma variedade de camadas predefinidas, incluindo convolucionais, recorrentes e densas.
* **Funções de ativação e otimização:** Keras fornece funções de ativação (por exemplo, ReLU, sigmoid) e otimizadores (por exemplo, Adam, RMSprop) para otimizar a rede neural.

**Funções de Perda e Métricas:**

Keras permite que os usuários definam funções de perda personalizadas para avaliar o desempenho do modelo. Algumas funções de perda comuns incluem:

* **Perda Quadrática Média (MSE):** MSE = 1/n Σ(yᵢ - ŷᵢ)²
* **Perda de Entropia Cruzada Binária (BCE):** BCE = -Σ[yᵢ \* log(ŷᵢ) + (1 - yᵢ) \* log(1 - ŷᵢ)]
* **Perda de Entropia Cruzada Categórica (CCE):** CCE = -Σ[yᵢ \* log(ŷᵢ)]

Keras também oferece uma ampla gama de métricas para avaliar os resultados do modelo, como precisão, recall e F1-score.

**Vantagens:**

* Fácil de usar e aprender
* Flexível e extensível
* Treinamento rápido e eficiente
* Ampla comunidade de suporte

**Aplicações:**

Keras é amplamente utilizado em uma variedade de aplicações de aprendizado de máquina, incluindo:

* Classificação de imagem
* Processamento de linguagem natural
* Detecção de objetos
* Série temporal