Gabriel Henrique Alves Spacki **10/12/2023**

Estatística para Aprendizado de Máquina

Ao lidar com análise de dados, é fundamental compreender a diversidade dos tipos de dados que podem ser encontrados. Essa categorização desempenha um papel crucial na interpretação e manipulação de informações, influenciando diretamente as abordagens analíticas apropriadas. Os principais tipos de dados incluem numéricos, que representam quantidades mensuráveis; categóricos, que expressam qualidades sem significado matemático intrínseco; e ordinais, uma combinação entre dados numéricos e categóricos, atribuindo significado matemático a categorias específicas. Essa classificação serve como alicerce essencial para a compreensão mais aprofundada das ferramentas e técnicas utilizadas na análise de dados.

Principais tipos de dados:

- **Numéricos:**
    - Representa algum tipo de medição quantitativa
        - Altura de pessoas
        - Tempo de carregamento de páginas
        - Preços de acções
    - Discretos
        - Baseado em números inteiros
        - Normalmente representam algum evento
            - Quantas compras um cliente fez durante um ano?
    - Contínuos
        - Possui um valor infinito de possibilidades
            - Quanto tempo demorou para um usuário realizar o check out?
            - Quanta chuva caiu em um determinado dia?

- **Categóricos:**
    - Dados qualitativos que não possuem significado matemático inerente
        - Gênero
        - Dados binários (sim/não)
        - Estado de residência
        - Categoria de um produto
    - Pode-se atribuir números às categorias para representá-las de forma mais compacta, mas os números não têm significado matemático

- **Ordinais:**
    - Uma mistura de dados numéricos com categóricos
    - Dados categóricos que possuem significado matemático
        - Avaliação de um filme em uma escala 1-5
            - Considerando que os valores possuem significado matemático
            - 1 significa ser pior que 2

Medidas de Tendência Central

- **Média:** Representa o valor médio de um conjunto de dados. Calcula-se somando todos os valores e dividindo pelo número de observações.
    - Exemplo: No conjunto {1, 2, 3, 4, 5}, a média é representada por (1 + 2 + 3 + 4 + 5) / 5, ou seja, 3.

- **Mediana:** É o valor central em um conjunto de dados ordenados. Em uma distribuição simétrica, a mediana é a mesma que a média.
    - Exemplo: No conjunto {1, 2, 3, 4, 5}, a mediana é representada pelo valor central, ou seja, 3.

- **Moda:** Refere-se ao valor mais frequente em um conjunto de dados. Pode haver uma moda, mais de uma ou nenhuma.
    - Exemplo: No conjunto {1, 1, 2, 2, 2, 3}, a moda é representada pelo número 2.

Medidas de Dispersão

- **Desvio Padrão:** Expressa a dispersão dos dados em relação à média. Valores mais altos indicam maior variabilidade.

- **Variância:** O quadrado do desvio padrão, fornece uma medida da dispersão dos dados ao quadrado.

As medidas de tendência central fornecem insights sobre a tendência central dos dados, enquanto as medidas de dispersão indicam o quão dispersos estão. Compreender esses conceitos é crucial para interpretar e analisar conjuntos de dados de forma eficaz.

Função Densidade de Probabilidade (FDP):

A FDP é usada em variáveis contínuas, modelando a probabilidade relativa de a variável assumir diferentes valores dentro de um intervalo. É expressa matematicamente e a integral sobre um intervalo fornece a probabilidade nesse intervalo. Exemplos incluem Distribuição Normal e Exponencial.

Função de Massa de Probabilidade (FMP):

Aplicada a variáveis discretas, a FMP atribui probabilidades a valores específicos. Usada para calcular a probabilidade de valores individuais em variáveis como o número de sucessos em experimentos discretos. A soma de todas as probabilidades é 1, refletindo certeza na observação de um valor discreto específico.

Funções de Distribuição de Dados:

É crucial compreender as funções de distribuição que modelam o comportamento de diferentes tipos de variáveis. Cada função oferece uma perspectiva única sobre como os dados estão distribuídos e fornece informações valiosas para tomadas de decisão e previsões.

- **Distribuição Uniforme:** Caracterizada por atribuir a mesma probabilidade a todos os valores possíveis dentro de um intervalo.
    - Exemplo: Ao lançar um dado justo, cada face tem a mesma probabilidade de ocorrer, resultando em uma distribuição uniforme discreta.

- **Distribuição Normal (ou Gaussiana):** Amplamente utilizada devido ao seu formato de sino e à propriedade de que a média, mediana e moda são iguais. Caracteriza-se por dois parâmetros: média (posição central da curva) e desvio padrão (dispersão dos dados em torno da média).
    - Muitos fenômenos naturais e sociais seguem essa distribuição, como a altura das pessoas.

- **Distribuição Exponencial:** Modela o tempo entre eventos em um processo de Poisson, onde eventos ocorrem de forma contínua e independente.
    - Amplamente aplicada em análise de confiabilidade e tempo de vida de produtos. Caracterizada por uma taxa de decaimento constante.

- **Distribuição Binomial:** Descreve o número de sucessos em uma série de tentativas independentes, cada uma com duas possíveis categorias (sucesso ou fracasso). Caracterizada por dois parâmetros: número de tentativas (n) e probabilidade de sucesso (p).
    - Exemplo: Lançamento repetido de uma moeda justa.

- **Distribuição de Poisson:** Modela o número de eventos que ocorrem em um intervalo fixo de tempo. Caracterizada por um único parâmetro, a taxa média de ocorrência de eventos. Aplicada em situações onde os eventos são raros, mas ocorrem de forma independente.

Visualização de dados

O matplotlib é uma biblioteca em Python para criação de gráficos e visualização de dados. Abaixo, seguem alguns dos principais tipos de gráficos que podem ser criados usando essa biblioteca:

- **Gráfico de Linhas:**
  - **plt.plot(x, y):** Exibe uma série de pontos conectados por linhas. Útil para representar tendências ao longo do tempo ou para mostrar a relação entre duas variáveis.

- **Gráfico de Dispersão:**
  - **plt.scatter(x, y):** Mostra pontos individuais em um plano cartesiano. Ótimo para identificar padrões, clusters ou outliers em conjuntos de dados.

- **Gráfico de Barras:**
  - **plt.bar(x, height):** Exibe barras verticais ou horizontais para representar a magnitude de diferentes categorias. Útil para comparações entre categorias discretas.

- **Gráfico de Barras Empilhadas:**
  - **plt.bar(x, y1, bottom=y2):** Semelhante ao gráfico de barras, mas as barras são empilhadas para representar a contribuição de diferentes partes para um todo.

- **Gráfico de Histograma:**
  - **plt.hist(data, bins):** Visualiza a distribuição de um conjunto de dados. Divide os dados em intervalos (blocos) e mostra a frequência de ocorrência em cada intervalo.

- **Gráfico de Pizza:**
  - **plt.pie(data, labels):** Representa a distribuição percentual de partes de um todo. Cada fatia do gráfico de pizza representa uma categoria e sua proporção em relação ao todo.

Seaborn é outra biblioteca poderosa em Python, construída sobre o Matplotlib, que simplifica ainda mais a criação de visualizações estatísticas atraentes. Foi projetada para trabalhar bem com estruturas de dados do tipo DataFrame do Pandas, tornando-a uma escolha popular para análise de dados e visualização em conjunto com Pandas. Entre os recursos oferecidos pelo Seaborn, podemos destacar:

- **Gráfico de Distribuição (Distplot):**
    - **seaborn.distplot(data):** Combina a funcionalidade de um histograma com uma estimativa de densidade kernel, fornecendo uma visão detalhada da distribuição dos dados.

- **Gráfico de Boxplot:**
    - **seaborn.boxplot(x, y, data):** Ótimo para visualizar a distribuição estatística dos dados, destacando a mediana, quartis e possíveis outliers.

- **Gráfico de Violino (Violin Plot):**
    - **seaborn.violinplot(x, y, data):** Combina um boxplot com uma estimativa da densidade de probabilidade nos lados, proporcionando uma visão mais completa da distribuição dos dados.

- **Mapa de Calor (Heatmap):**
    - **seaborn.heatmap(data):** Ideal para representar matrizes de dados, destacando padrões e correlações entre variáveis.

- **Gráfico de Regressão:**
    - **seaborn.regplot(x, y, data):** Facilita a visualização de relações lineares entre variáveis, incluindo uma linha de regressão.

Em resumo, compreender a diversidade dos tipos de dados é crucial para uma análise de dados eficaz. A categorização de dados fornece a base necessária para a escolha adequada de ferramentas e técnicas. As medidas de tendência central e dispersão oferecem insights valiosos sobre a distribuição dos dados. Na visualização de dados, tanto o Matplotlib quanto o Seaborn oferecem opções variadas para explorar padrões. Em síntese, a combinação de fundamentos estatísticos sólidos e ferramentas de visualização robustas é essencial para uma análise de dados abrangente.