E-BOOK
DO ESTUDANTE

Estatística

Além dos Números

RESILIA

SUMÁRIO

Estatística

Além dos Números

©pixabay

A Estatística é uma área da Matemática que se dedica à coleta, análise e interpretação de dados numéricos. Ela é utilizada em diversas áreas do conhecimento, como na saúde, economia, ciência política, psicologia, entre outras.

A Estatística é composta por duas áreas principais: a Estatística Descritiva e a Estatística Inferencial. A Estatística Descritiva se dedica a descrever e resumir as características dos dados de forma objetiva, utilizando medidas como a média, mediana, moda e desvio-padrão. Já a Estatística Inferencial utiliza técnicas para inferir informações sobre uma população a partir de uma amostra dos dados.

Para um profissional analista de dados, a estatística é essencial, pois ela oferece as ferramentas necessárias para coletar e analisar dados, testar hipóteses e fazer previsões precisas e confiáveis. Dessa forma, neste material, daremos informações sobre o fundamento de Estatística e de Estatística Descritiva, fazendo uso da linguagem Python para explicar sua aplicação.

CONTEXTUALIZANDO

A Estatística é uma habilidade fundamental para qualquer profissional analista de dados, pois permite a coleta, a análise e a interpretação de dados de maneira confiável e eficaz para a tomada de decisões informada e baseada em evidências. É uma ferramenta importante ainda para tomar decisões com base em fatos, tanto em ciências exatas como em ciências sociais, negócios e finanças.

Python é uma linguagem que oferece vários recursos para se trabalhar com Estatística. Essa linguagem possui diversas bibliotecas específicas para tal, como Numpy, Pandas, Matplotlib e Scipy. Essas bibliotecas facilitam a manipulação e a visualização de dados, além de permitir a realização de cálculos estatísticos complexos.

No mercado de trabalho, a Estatística é cada vez mais valorizada em diversas áreas, como finanças, marketing, saúde e tecnologia, dentre outras. Dessa forma, é relevante que o analista de dados tenha uma boa base em Estatística para que consiga trabalhar corretamente com os dados e, assim, gerar insights importantes para os negócios em que atua.

CONCEITOS FUNDAMENTAIS DE ESTATÍSTICA

A Estatística é uma disciplina da Matemática que trata do estudo de dados e de informações para extrair conclusões e inferências sobre determinado fenômeno. O analista de dados precisa conhecer os principais conceitos da Estatística para ser ágil e preciso em pré-processamento de dados e realizar conclusões e tomada de decisões estratégicas, dentre outras operações importantes.

A **Estatística Descritiva** é uma parte da Estatística que se preocupa com a descrição e a apresentação dos dados. É uma das primeiras etapas na análise de dados, e ajuda a compreender a natureza e as características dos dados coletados. Algumas das medidas descritivas mais comuns incluem a média, a mediana, a moda, a variância e o desvio-padrão.

Seguem alguns exemplos de como utilizar a linguagem Python para realizar análises estatísticas descritivas.

Para começar, vamos importar a biblioteca Pandas e carregar um arquivo CSV contendo dados em um arquivo .csv qualquer:

```
import pandas as pd

vendas = pd.read_csv('arquivo.csv')
```

Podemos utilizar o método head() para visualizar as primeiras linhas do conjunto de dados. Isso pode ser útil caso você queira traçar uma linha geral do que seu arquivo possui, ou ir observando alguma recorrência nos dados.

```
print(vendas.head())
```

Média é uma medida estatística que representa o valor central de um conjunto de dados. Ela é obtida somando todos os valores e os dividindo pela quantidade total de elementos, e é útil para identificar tendências e padrões em um conjunto de dados.

Por exemplo, se uma empresa quer calcular a média salarial dos seus funcionários, basta somar o salário de cada um e dividir pelo número total de funcionários. Veja um exemplo diferente usando Python. Podemos calcular a média das vendas utilizando o método mean():

```
media_vendas = vendas['valor'].mean()
print('Média das vendas: ', media_vendas)
```

Já a **mediana** é a medida estatística que representa o valor que separa os dados em duas partes iguais. Para encontrá-la, é necessário ordenar os dados e identificar o valor central. A mediana é útil para identificar a distribuição de dados em um conjunto.

Por exemplo, se uma empresa quer saber qual é a mediana de idade dos seus funcionários, basta ordenar as idades do mais novo para o mais velho e encontrar o valor central. Veja um exemplo em Python. Para calcular a mediana das vendas, podemos utilizar o método median():

```
mediana_vendas = vendas['valor'].median()
print('Mediana das vendas: ', mediana_vendas)
```

A **moda** é outra medida estatística que representa o valor mais frequente em um conjunto de dados. Em outras palavras, é o valor que aparece com mais frequência. A moda é útil para identificar padrões e preferências em um conjunto de dados.

Por exemplo, se uma empresa quer saber qual é o modelo de carro mais vendido em uma determinada região, a moda seria o modelo que aparece com mais frequência. Veja um exemplo em Python. Para calcular a moda das vendas, podemos utilizar o método mode():

```
moda_vendas = vendas['valor'].mode()
print('Moda das vendas: ', moda_vendas)
```

A **variância** é uma medida estatística que mede o quão dispersos estão os valores de um conjunto de dados em relação à média. Em outras palavras, ela indica a variabilidade dos dados em torno da média. A variância é calculada pela média dos quadrados das diferenças entre cada valor do conjunto de dados e a média do conjunto.

Uma variância alta indica que os dados estão muito espalhados, enquanto uma variância baixa indica que os dados estão mais próximos da média. A variância é amplamente utilizada na estatística e em outras áreas, como finanças e engenharia, para medir a incerteza e a variabilidade dos dados, bem como para avaliar a precisão e a confiabilidade de modelos e previsões.

Veja um exemplo em Python. Podemos calcular a variância das vendas utilizando o método var():

```
variancia_vendas = vendas['valor'].var()
print('Variância das vendas: ', variancia_vendas)
```

Desvio-padrão é a medida estatística que representa a variação dos dados em relação à média. Ele mede o quanto os dados estão afastados da média e ajuda a entender a dispersão dos dados em um conjunto. O desvio-padrão é útil para avaliar a homogeneidade dos dados em um conjunto.

Por exemplo, se uma empresa quer saber o desvio-padrão do salário dos seus funcionários, ela pode identificar o quanto os salários se afastam da média salarial. Veja um exemplo em Python. Para calcular o desvio-padrão das vendas, podemos utilizar o método std():

```
desvio_padrao_vendas = vendas['valor'].std()
print('Desvio padrão das vendas: ', desvio_padrao_vendas)
```

Vimos alguns conceitos fundamentais relacionados à Estatística e suas aplicações usando a linguagem Python. No tópico a seguir, veremos mais dois conceitos dessa natureza, mas com um grau de complexidade maior. Então, vamos lá!

CORRELAÇÃO E COVARIÂNCIA

A **covariância** e a **correlação** são dois conceitos fundamentais na análise estatística que nos ajudam a entender a relação entre duas variáveis aleatórias. Ambas as medidas indicam o grau de **associação linear** entre as duas variáveis, mas de formas diferentes.

Glossário: Associação linear refere-se a um tipo de relação entre duas variáveis quantitativas, na qual a relação entre elas pode ser descrita por uma linha reta. Em outras palavras, a associação linear significa que, à medida que o valor de uma variável aumenta ou diminui, o valor da outra variável também aumenta ou diminui de maneira proporcional.

A **covariância** é uma medida bruta que nos diz como as duas variáveis estão relacionadas em termos de unidades de medida, enquanto a **correlação** é uma medida padronizada que nos indica como as duas variáveis estão relacionadas independentemente de suas unidades de medida.

A fórmula da **covariância** é uma medida estatística que indica o grau de associação linear entre duas variáveis aleatórias. É calculada pela média dos produtos das diferenças de cada valor da variável em relação à média de sua respectiva variável. Em outras palavras, ela mede como as duas variáveis variam juntas. A fórmula da covariância é:

Covariância(X,Y) = (1 / n-1) * somatório de (Xi - Xmédio) * (Yi - Ymédio)

Onde X e Y são as duas variáveis aleatórias, Xi e Yi são as observações correspondentes a cada variável, Xmédio e Ymédio são as médias das duas variáveis e n é o número de observações.

Se a covariância for **positiva**, isso indica que as duas variáveis estão associadas positivamente, ou seja, quando uma variável aumenta, a outra também aumenta. Se a covariância for negativa, isso indica que as duas variáveis estão associadas negativamente, ou seja, quando uma variável aumenta, a outra diminui. Por outro lado, se a covariância for **zero**, isso indica que as duas variáveis são independentes, ou seja, uma variável não afeta a outra.

Podemos calcular a covariância em Python usando a função cov() da biblioteca NumPy. Vamos ver um exemplo:

```
import numpy as np

# Definir duas variáveis
x = [1, 2, 3, 4, 5]
y = [3, 5, 7, 9, 11]

# Calcular a covariância
cov = np.cov(x, y)[0][1]
print("Covariância: ", cov)
```

Nesse exemplo, definimos duas variáveis x e y. Em seguida, usamos a função np.cov() para calcular a covariância entre as duas variáveis. O resultado da covariância é 8.5, o que indica uma associação positiva entre as duas variáveis.

A **correlação** é uma medida estatística que indica o grau de associação linear entre duas variáveis aleatórias, mas ela é calculada dividindo a covariância pelo produto dos desvios-padrão das duas variáveis. A correlação é uma medida padronizada que varia entre **-1 e 1**, onde 1 indica uma **correlação positiva perfeita**, -1 indica uma **correlação negativa perfeita** e 0 indica que **não há correlação**.

A fórmula da correlação é:

Correlação(X,Y) = Covariância(X,Y) / (desvio padrão de X * desvio padrão de Y)

Para calcular a correlação em Python, podemos utilizar a função "corr()" da biblioteca "pandas".

Veja um exemplo de como calcular a correlação entre duas variáveis em Python:

```
import pandas as pd

# Cria um DataFrame com duas variáveis
df = pd.DataFrame({'variavel1': [1, 2, 3, 4, 5], 'variavel2':
[10, 20, 30, 40, 50]})

# Calcula a correlação entre as variáveis
correlacao = df['variavel1'].corr(df['variavel2'])

print('A correlação entre as variáveis é:', correlacao)
```

Nesse exemplo, a correlação entre as variáveis é de 1, o que indica uma correlação positiva perfeita entre elas.

Assim, vimos as definições de covariância e correlação bem como a forma com a qual podemos desenvolver códigos na linguagem python usando esses recursos da estatística.

Para refletir

- Por que um analista de dados precisa ter conhecimento sobre Estatística?
- Como a linguagem Python pode ajudar na análise estatística de dados?
- Quais são as principais habilidades em Estatística que as empresas procuram em seus candidatos a analistas de dados?

Atividade: Estatística e Python

Que tal nos aprofundarmos um pouco mais nesse tema de maneira prática? Baixe o arquivo "students_performance.csv" disponível no link: https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.csv . Esse arquivo contém dados sobre as pontuações dos alunos em exames em diferentes áreas.
Utilize a biblioteca Pandas para calcular as seguintes estatísticas descritivas: média, moda, mediana, desvio padrão, correlação e covariância.
Siga os passos abaixo:

- Importe a biblioteca pandas;
- Leia o arquivo "students_performance.csv" utilizando a função "read_csv" do pandas e atribua-o a uma variável;
- Calcule a média das pontuações de matemática utilizando a função "mean" do pandas;
- Calcule a moda das pontuações de leitura utilizando a função "mode" do pandas
- Calcule a mediana das pontuações de escrita utilizando a função "median" do pandas;
- Calcule o desvio-padrão das pontuações de matemática utilizando a função "std" do pandas;
- Calcule a correlação entre as pontuações de matemática e de leitura utilizando a função "corr" do pandas;
- Calcule a covariância entre as pontuações de matemática e de escrita utilizando a função "cov" do pandas.

Para ir além

- Vimos, neste material, alguns conceitos fundamentais sobre Estatística. Nos sites abaixo, descubra mais sobre média, moda e mediana usando a linguagem python.
<http://www.bosontreinamentos.com.br/programacao-em-python/medidas-de-tendencia-central-media-moda-e-mediana-em-python/>
<https://www.pythonprogressivo.net/2018/02/Calculando-Media-Aritmetica-Python-Precedencia-Operadores.html>

- Vimos que correlação é uma medida estatística que indica a relação entre duas variáveis. No site abaixo, encontre algumas aplicações de correlação usando a linguagem python.
<https://medium.com/omixdata/estat%C3%ADstica-an%C3%A1lise-de-correla%C3%A7%C3%A3o-usando-python-e-r-d68611511b5a>

- Vimos que a covariância é uma medida estatística que indica o grau de interdependência entre duas variáveis aleatórias. No site abaixo, descubra algumas implementações de correlação usando a linguagem python.
<https://www.delftstack.com/pt/howto/numpy/python-covariance/>

RESUMO DE ESTATÍSTICA

CONCEITOS FUNDAMENTAIS DE ESTATÍSTICA
A linguagem Python é uma linguagem de programação que tem se tornado cada vez mais popular no campo da ciência de dados e da análise estatística.

Média

```
media_vendas = vendas['valor'].mean()
print('Média das vendas: ', media_vendas)
```

Mediana

```
mediana_vendas = vendas['valor'].median()
print('Mediana das vendas: ', mediana_vendas)
```

Moda

```
moda_vendas = vendas['valor'].mode()
print('Moda das vendas: ', moda_vendas)
```

Variância

```
variancia_vendas = vendas['valor'].var()
print('Variância das vendas: ',
variancia_vendas)
```

Desvio Padrão

```
desvio_padrao_vendas = vendas['valor'].std()
print('Desvio padrão das vendas: ',
desvio_padrao_vendas)
```

CORRELAÇÃO E COVALÊNCIA
A covariância e a correlação são dois conceitos fundamentais na análise estatística que nos ajudam a entender a relação entre duas variáveis aleatórias.

Correlação

```
import numpy as np

# Definir duas variáveis
x = [1, 2, 3, 4, 5]
y = [3, 5, 7, 9, 11]

# Calcular a covariância
cov = np.cov(x, y)[0][1]
print("Covariância: ", cov)
```

Covariância

```
import pandas as pd

# Cria um DataFrame com duas variáveis
df = pd.DataFrame({'variavel1': [1, 2, 3, 4,
5], 'variavel2': [10, 20, 30, 40, 50]})

# Calcula a correlação entre as variáveis
correlacao = df['variavel1'].corr(df
['variavel2'])

print('A correlação entre as variáveis é:',
correlacao)
```

Até a próxima e
#confianoprocesso

RESILIA