

**Análise Dataset de**

**Leitos 2024**

**Estatística**

**Adriana Rodrigues**

**Lorena Seabra**

**2024.1Relatório Técnico:**

**Análise do Dataset de Leitos 2024**

Este relatório técnico apresenta uma análise do dataset **Leitos2024**, que contém informações sobre a disponibilidade de leitos e UTIs (Unidades de Terapia Intensiva) no Brasil no período de janeiro a março de 2024. Os dados incluem tanto os leitos e UTIs **existentes** em geral quanto os **disponibilizados** pelo Sistema Único de Saúde (SUS).

Fonte: <https://opendatasus.saude.gov.br/pt_BR/dataset/hospitais-e-leitos>

# **1. Estrutura do Dataset**

O dataset original possui as seguintes colunas:

* **COMP**: Competência (período de referência dos dados)
* **REGIAO**: Região geográfica do Brasil
* **UF**: Unidade Federativa (Estado)
* **MUNICIPIO**: Município
* **NOME\_ESTABELECIMENTO**: Nome do estabelecimento de saúde
* **TP\_GESTAO**: Tipo de gestão (municipal, estadual, federal)
* **DS\_TIPO\_UNIDADE**: Descrição do tipo de unidade
* **DESC\_NATUREZA\_JURIDICA**: Natureza jurídica do estabelecimento
* **LEITOS\_EXISTENTES**: Total de leitos existentes
* **LEITOS\_SUS**: Total de leitos disponibilizados pelo SUS
* **UTI\_TOTAL\_EXIST**: Total de UTIs existentes
* **UTI\_TOTAL\_SUS**: Total de UTIs disponibilizadas pelo SUS
* **UTI\_ADULTO\_EXIST**: Total de UTIs para adultos existentes
* **UTI\_ADULTO\_SUS**: Total de UTIs para adultos disponibilizadas pelo SUS
* **UTI\_PEDIATRICO\_EXIST**: Total de UTIs pediátricas existentes
* **UTI\_PEDIATRICO\_SUS**: Total de UTIs pediátricas disponibilizadas pelo SUS
* **UTI\_NEONATAL\_EXIST**: Total de UTIs neonatais existentes
* **UTI\_NEONATAL\_SUS**: Total de UTIs neonatais disponibilizadas pelo SUS
* **UTI\_QUEIMADO\_EXIST**: Total de UTIs para queimados existentes
* **UTI\_QUEIMADO\_SUS**: Total de UTIs para queimados disponibilizadas pelo SUS
* **UTI\_CORONARIANA\_EXIST**: Total de UTIs coronarianas existentes
* **UTI\_CORONARIANA\_SUS**: Total de UTIs coronarianas disponibilizadas pelo SUS

# **2. Metodologia**

O relatório técnico foi iniciado com a análise das distribuições de leitos e UTIs por região e unidade federativa (UF). Utilizamos técnicas de **Estatística Descritiva** e **Visualização de Dados**, incluindo:

**Estatística Descritiva**

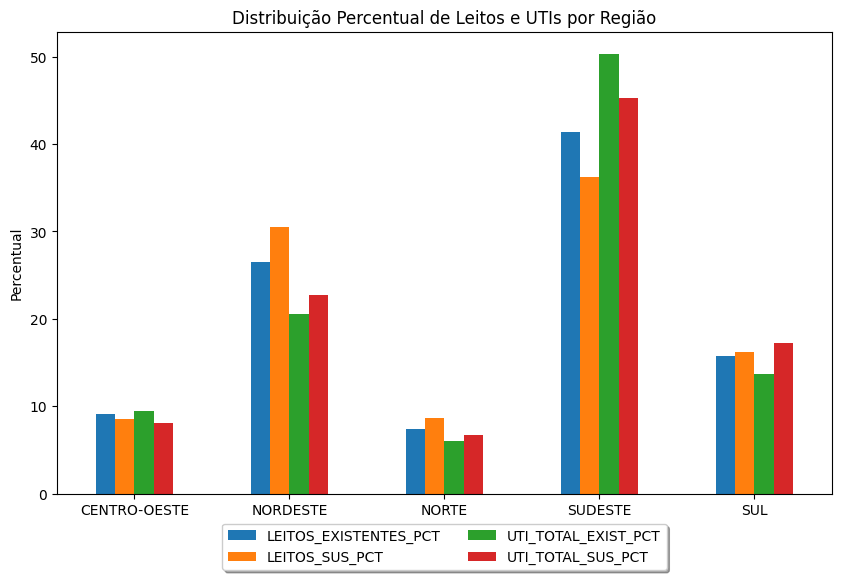
1. Estatística Descritiva.
2. Normal Padrão.
3. Distribuições Amostrais.
4. Intervalo de Confiança.

**Visualização de Dados**

1. **Box plot**: Para análise de dispersão e identificação de outliers.
2. **Histograma**: Para visualizar a distribuição dos dados.
3. **Scatter plot**: Para explorar relações entre variáveis.
4. **Mapa de calor (heatmap)**: Para correlação entre variáveis.
5. **Distribuições Amostrais**: Análise de distribuição dos dados.
6. **Intervalo de Confianç**a: Para inferências estatísticas.

# 

# **3. Distribuição de Leitos Existentes, Leitos SUS, UTI Existentes e UTI do SUS por Região e UF**



REGIAO EITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS

CENTRO-OESTE 9.078810 8.521817 9.493742 8.128393

NORDESTE 26.456291 30.513230 20.548406 22.679655

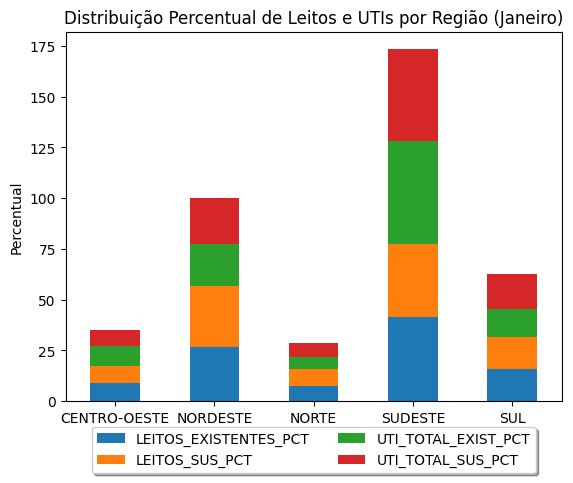
NORTE 7.403277 8.613497 5.982468 6.697541

SUDESTE 41.338506 36.205466 50.337345 45.292239

SUL 15.723118 16.145990 13.638039 17.202172

A **região Sudeste** possui a maior proporção de leitos e de UTI existentes e também a maior proporção de leitos do SUS. A **região Nordeste** tem a segunda maior proporção de leitos e de UTI, porém possui maior proporção de leitos do SUS em relação a leitos existentes. Já **região Sul** com a terceira maior proporção de leitos e de UTI, porém com maior proporção de unidades do SUS. Em seguida, vêm as regiões **Centro-Oeste** e **Norte** com as menores proporção de leitos e de UTI.

## 



REGIAO EITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS

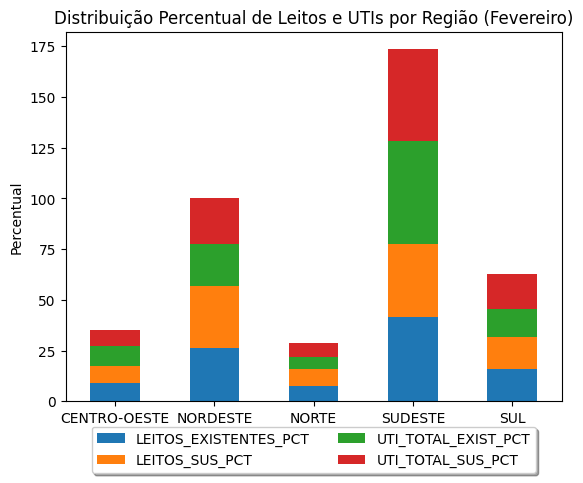
CENTRO-OESTE 9.059410 8.492359 9.490071 8.191181

NORDESTE 26.491120 30.510383 20.609215 22.635983

NORTE 7.387497 8.606133 5.929889 6.736402

SUDESTE 41.398206 36.285826 50.454662 45.259092

SUL 15.663767 16.105298 13.516162 17.177341



REGIAO EITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS

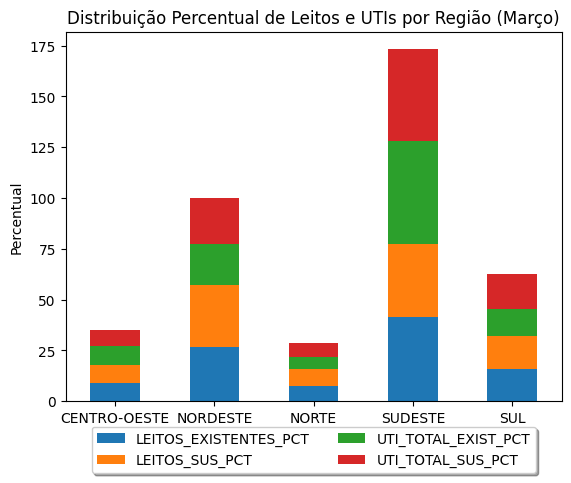
CENTRO-OESTE 9.044688 8.510206 9.458833 8.191445

NORDESTE 26.458830 30.513901 20.517332 22.636712

NORTE 7.375104 8.606195 5.928367 6.736618

SUDESTE 41.437118 36.268571 50.499896 45.254112

SUL 15.684259 16.101127 13.595572 17.181113



REGIAO EITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS

CENTRO-OESTE 9.078810 8.521817 9.493742 8.128393

NORDESTE 26.456291 30.513230 20.548406 22.679655

NORTE 7.403277 8.613497 5.982468 6.697541

SUDESTE 41.338506 36.205466 50.337345 45.292239

SUL 15.723118 16.145990 13.638039 17.202172

Nos gráficos por mês podemos observar algumas tendências na distribuição de leitos existentes, leitos SUS e leitos de UTI.

Na **região Centro-Oeste** os **leitos existentes** tiveram um pequeno aumento de janeiro para fevereiro, mas permaneceram estáveis em março. Os **leitos SUS** também tiveram um aumento discreto de janeiro para fevereiro, seguido de uma pequena queda em março. A **porcentagem de UTI total SU**\* se manteve relativamente constante ao longo dos três meses.

Na região **Nordeste** os **leitos existentes** tiveram uma pequena queda de janeiro para fevereiro, mas se recuperaram em março. Os **leitos SUS** tiveram um aumento significativo de janeiro para fevereiro, mas diminuíram em março. A **porcentagem de UTI total SUS** também apresentou variações, mas permaneceu relativamente estável.

Na **região Norte** os **leitos existentes** tiveram uma pequena variação, mas permaneceram próximos em todos os meses. Os **leitos SUS** também tiveram pouca variação. A **porcentagem de UTI total SUS** se manteve constante.

Na **região Sudeste** os **leitos existentes** tiveram uma queda significativa de janeiro para fevereiro, mas se recuperaram em março. Os **leitos SUS** tiveram uma variação semelhante, com queda em fevereiro e aumento em março. A **porcentagem de UTI total SUS** permaneceu relativamente estável.

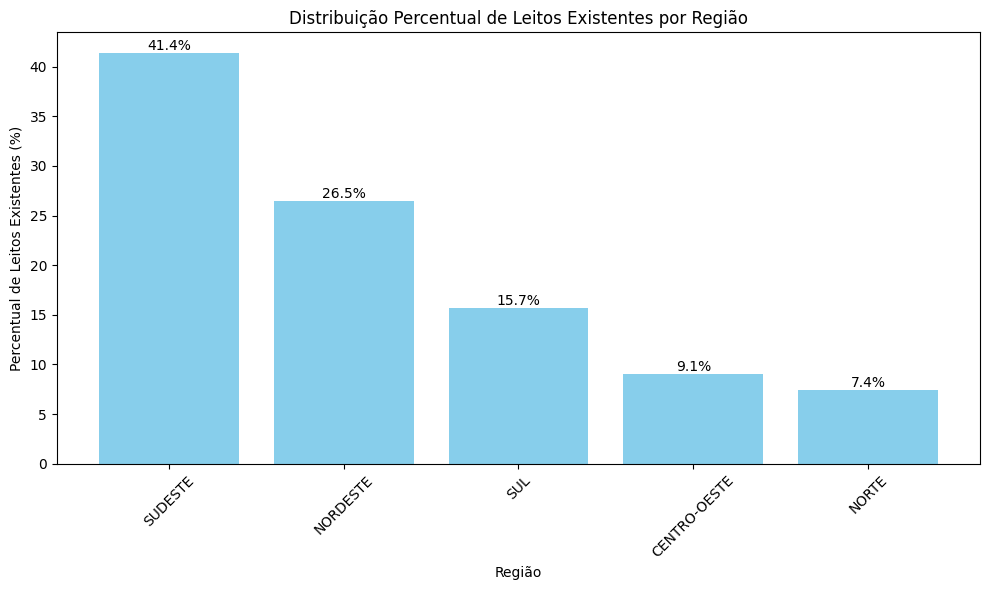
Na **região Sul** os **leitos existentes** tiveram uma pequena variação, mas permaneceram próximos em todos os meses. Os **leitos SUS** também tiveram pouca variação. A **porcentagem de UTI total SUS** se manteve constante.

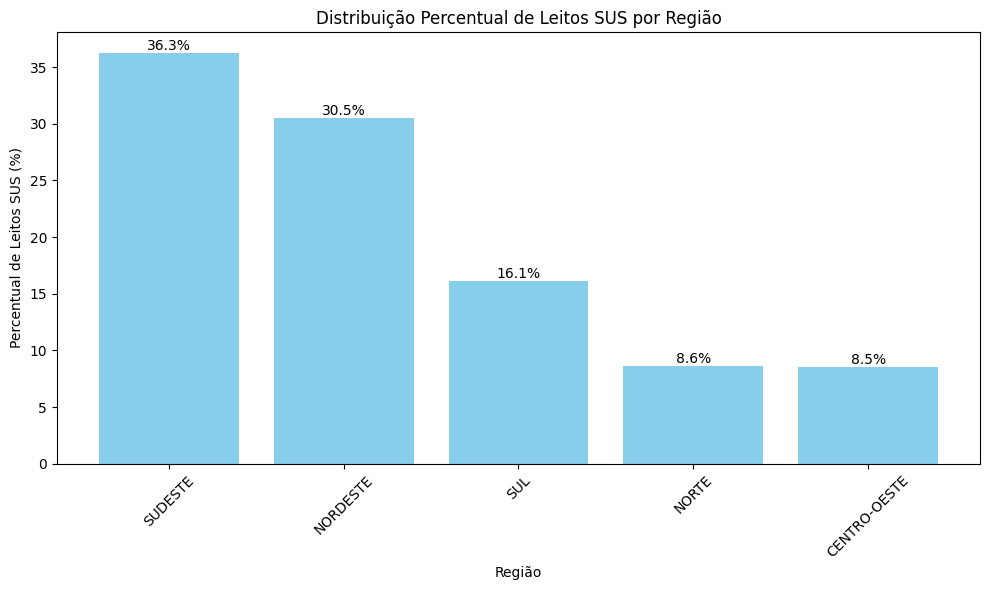
O **Nordeste** foi a região que teve as maiores variações nos leitos SUS, enquanto a região **Sudeste** teve as maiores variações nos leitos existentes. A porcentagem de UTI total SUS se manteve relativamente estável em todas as regiões.

## **3.1. Por região**

### **3.1.1. Leitos Existentes e Leitos SUS**

A distribuição de leitos existentes e leitos do Sistema Único de Saúde (SUS) varia significativamente entre as diferentes regiões do Brasil.





A **região Sudeste** possui a maior proporção de leitos existentes (41.39%) e também a maior proporção de leitos do SUS (36.25%). Isso sugere que, apesar de ter uma infraestrutura hospitalar robusta, ainda há uma parcela significativa de leitos destinados ao SUS.

A **região Nordeste** tem a segunda maior proporção de leitos existentes (26.47%) e a segunda maior proporção de leitos do SUS (30.51%). Essa região enfrenta desafios em termos de infraestrutura, mas está trabalhando para garantir o acesso aos serviços de saúde.

A **região Sul** apresenta uma proporção de leitos existentes de 15.69% e uma proporção de leitos do SUS de 16.12%. Esses percentuais são relativamente próximos, o que indica que a região Sul está comprometida em oferecer tanto leitos do SUS quanto leitos não vinculados ao sistema público de saúde.

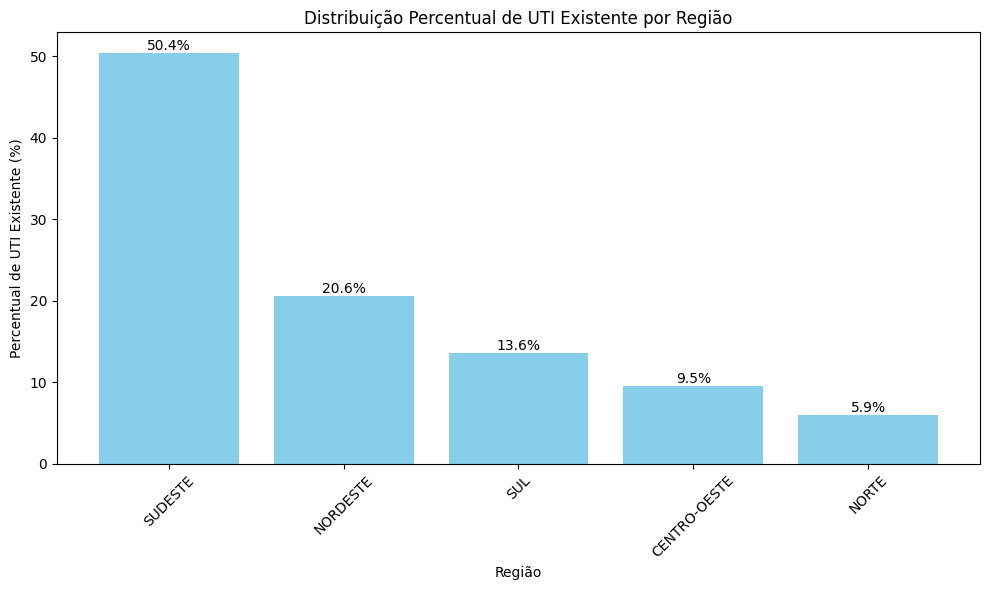
A **região Centro-Oeste** apresenta 9.06% dos leitos existentes e uma proporção de leitos do SUS de 8,05%. Essa região concentra uma quantidade relativamente menor de leitos e isso pode indicar desafios na infraestrutura de saúde nessa área.

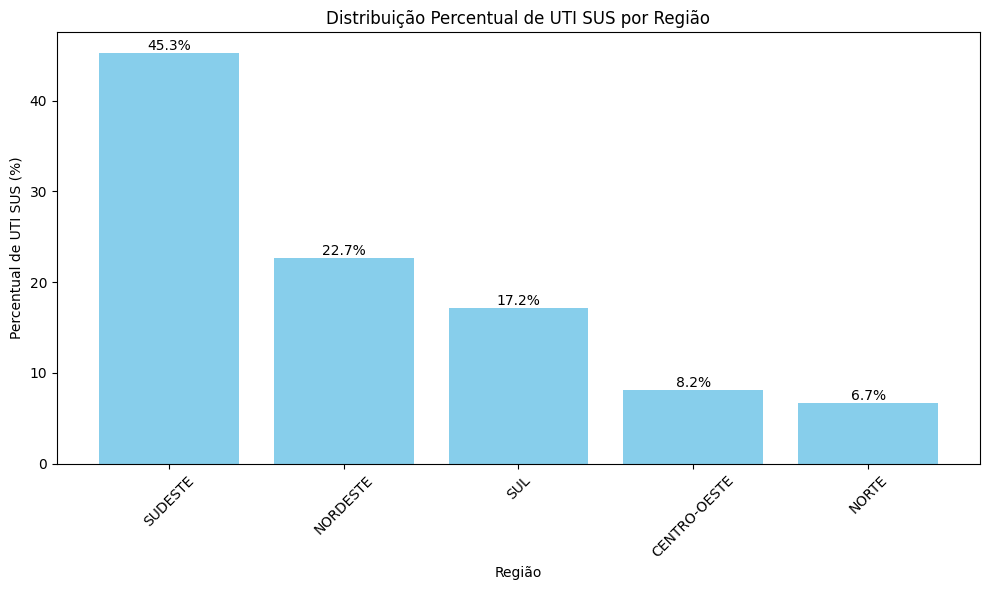
Embora seja uma proporção relativamente baixa, é importante considerar que essa região abrange estados como Goiás, Mato Grosso, Mato Grosso do Sul e o Distrito Federal. O Distrito Federal, em particular, concentra uma infraestrutura hospitalar significativa, o que pode influenciar esse percentual.

A **região Norte** a tem a menor proporção de leitos existentes (7.39%) e a menor proporção de leitos do SUS (8.61%). Esta região abrange estados com características geográficas e populacionais diversas. Mas, esses percentuais refletem a necessidade de investimentos contínuos para expandir a capacidade hospitalar na região.

### **3.1.2. UTI Existentes e UTI SUS**

A distribuição de UTI existentes e UTI do SUS varia consideravelmente entre as diferentes regiões do Brasil.



A **região Sudeste** possui a maior proporção de leitos existentes (41.39%) e também a maior proporção de leitos do SUS (36.25%). Esses números indicam que, apesar de ter uma infraestrutura hospitalar robusta, ainda há uma parcela significativa de leitos destinados ao SUS.

A **região Nordeste** tem a segunda maior proporção de leitos existentes (26.47%) e a segunda maior proporção de leitos do SUS (30.51%). Essa região abrange estados com características diversas, e a infraestrutura de saúde pode variar consideravelmente entre eles.

A **região Sul** apresenta uma proporção de leitos existentes de 15.69% e uma proporção de leitos do SUS de 16.12%. Esses percentuais são relativamente próximos, indicando um compromisso com a saúde pública.

A **região Centro-Oeste** concentra 9.06% dos leitos existentes e uma proporção de leitos do SUS de 8.05%. A proporção de leitos e UTIs do SUS é menor, indicando a necessidade de investimentos contínuos.

A **região Norte** tem a menor proporção de leitos existentes (7.39%) e a menor proporção de leitos do SUS (8.61%). Esses percentuais refletem a necessidade de investimentos contínuos para expandir a capacidade hospitalar na região, que abrange estados com características geográficas e populacionais diversas.

Fazendo um comparativo da disponibilidade de leitos e UTI vs. quantidade de habitantes por região, temos os seguintes insights:

Segundo o último censo do IBGE, a região Nordeste possui aproximadamente 53.081.950 habitantes. Isso nos leva a calcular a estimativa de leitos por habitante:

A **região Nordeste** possui um total de 134.971 leitos hospitalares existentes, dos quais 105.173 são no SUS. Além disso, existem 12.822 leitos de UTI no total, sendo 7.101 no SUS. Isso resulta em uma m**édia de 3,59 leitos por 10 mil habitantes**. Analisando separadamente, os números do SUS apresentam uma média de **1,28 leitos por 10 mil habitantes**, enquanto a rede privada tem **2,31 leitos por 10 mil habitantes**.

A **região Sudeste** com uma população estimada em 84.847.187 habitantes, possui um total de 210.895 leitos hospitalares existentes, dos quais 124.793 são destinados ao SUS. Além disso, existem 31.410 leitos de UTI no total, sendo 14.181 no SUS. Isso resulta em uma m**édia de 24,88 leitos por 10 mil habitantes na região Sudeste**.Analisando separadamente, os números do SUS apresentam uma **média de 14,73 leitos por 10 mil habitantes**, enquanto a rede privada tem uma **média de 10,15 leitos por 10 mil habitantes**.

A **região Sul** com uma população estimada em 29.933.315 habitantes, possui um total de 80.214 leitos hospitalares existentes, dos quais 55.652 são destinados ao SUS. Além disso, existem 8.510 leitos de UTI no total, sendo 5.386 no SUS. Isso resulta em uma **média de 26,80 leitos por 10 mil habitantes na região Sul**. Analisando separadamente, os números do SUS apresentam uma **média de 18,60 leitos por 10 mil habitantes**, enquanto a rede privada tem uma **média de 7,20 leitos por 10 mil habitantes**.

A **região Centro-Oeste** com uma população estimada em 16.287.809 habitantes, possui um total de 46.317 leitos hospitalares existentes, dos quais 29.373 são destinados ao SUS. Além disso, existem 5.924 leitos de UTI no total, sendo 2.545 no SUS. Isso resulta em uma **média de 28,41 leitos por 10 mil habitantes na região Centro-Oeste**. Analisando separadamente, os números do SUS apresentam uma **média de 18,04 leitos por 10 mil habitantes**, enquanto a rede privada tem uma **média de 10,37 leitos por 10 mil habitantes**.

A **região Norte** com uma população estimada em 17.349.619 habitantes, possui um total de 37.769 leitos hospitalares existentes, dos quais 29.689 são destinados ao SUS. Além disso, existem 3.733 leitos de UTI no total, sendo 2.097 no SUS. Isso resulta em uma **média de 21,77 leitos por 10 mil habitantes na região Norte**. Analisando separadamente, os números do SUS apresentam uma **média de 17,12 leitos por 10 mil habitantes**, enquanto a rede privada tem uma **média de 4,65 leitos por 10 mil habitantes**.

Essas análises refletem os totais de leitos e UTI, com base na população estimada em cada região. E esses dados são essenciais para compreender a capacidade de atendimento hospitalar na região

**Conclusão:**

Com base na análise da disponibilidades de leitos no mês de março/2024, podemos concluir que a região Nordeste, apesar de ter uma quantidade significativa de leitos e UTIs existentes e destinados ao SUS, apresenta uma média relativamente baixa de leitos por habitante. Essa média é inferior à das outras regiões, como Centro-Oeste e Norte.

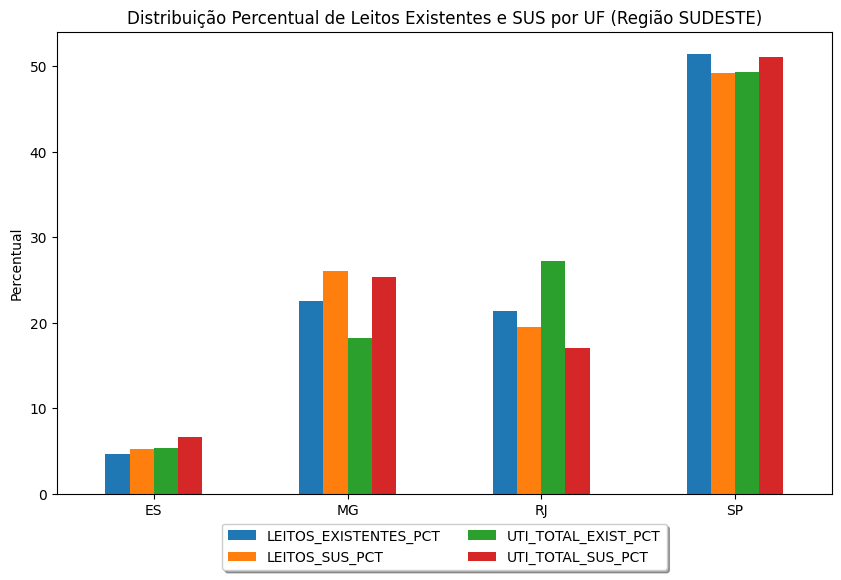
A distribuição desigual de leitos e UTIs impactar no acesso aos serviços de saúde, especialmente em situações de alta demanda, como epidemias ou emergências. Por isso, é muito importante que as autoridades de saúde avaliem esses dados e busquem estratégias para melhorar a distribuição e garantir um atendimento adequado à população.

**Fonte da quantidade de habitantes por Região/UF:**

[**IBGE**](https://www.ibge.gov.br/apps/populacao/projecao/)[**https://www.ibge.gov.br/apps/populacao/projecao/**](https://www.ibge.gov.br/apps/populacao/projecao/)

## **3.2. Por UF**

### **3.2.1. Leitos e UTI existentes e do SUS**



**UF LEITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS**

ES 4.629165 5.219270 5.344126 6.616552

MG 22.500655 26.089415 18.163036 25.324445

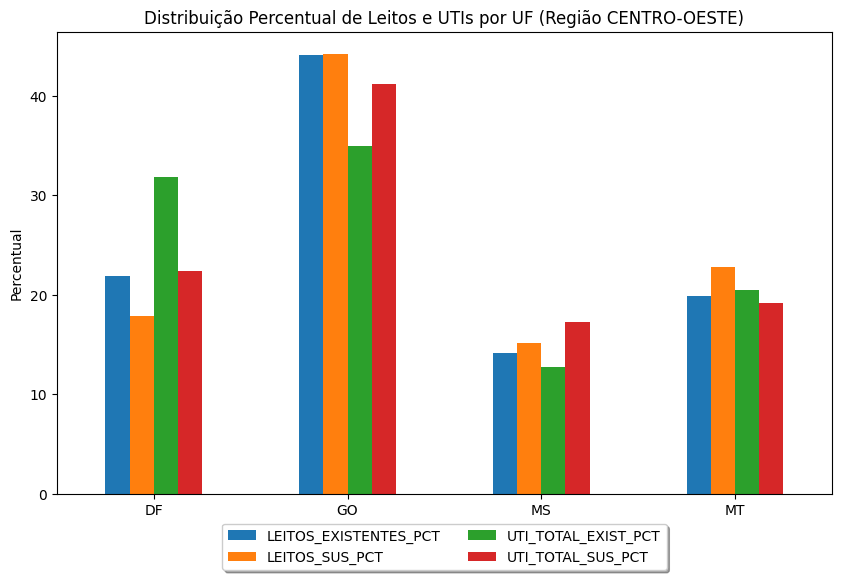
RJ 21.411644 19.502774 27.235548 17.017706

SP 51.458536 49.188540 49.257289 51.041297

Com base nos dados dos percentuais de leitos e UTIs por UF na região Sudeste, podemos fazer as seguintes análises:

1. **Espírito Santo (ES)**:
   * **Leitos existentes**: Representam cerca de 4,63% do total da região Sudeste.
   * **Leitos SUS**: Correspondem a aproximadamente 5,22% do total.
   * **UTI total existente**: Representa cerca de 5,34%.
   * **UTI total SUS**: É responsável por aproximadamente 6,62%.
2. **Minas Gerais (MG)**:
   * **Leitos existentes**: MG possui a maior parcela de leitos existentes na região, com cerca de 22,50%.
   * **Leitos SUS**: Representam aproximadamente 26,09%.
   * **UTI total existente**: Corresponde a cerca de 18,16%.
   * **UTI total SUS**: MG também lidera em UTIs SUS, com aproximadamente 25,32%.
3. **Rio de Janeiro (RJ)**:
   * **Leitos existentes**: RJ tem uma parcela significativa de leitos existentes, com cerca de 21,41%.
   * **Leitos SUS**: Representam aproximadamente 19,50%.
   * **UTI total existente**: Corresponde a cerca de 27,24%.
   * **UTI total SUS**: É responsável por aproximadamente 17,02%.
4. **São Paulo (SP)**:
   * **Leitos existentes**: SP lidera em leitos existentes, com cerca de 51,46% do total da região.
   * **Leitos SUS**: Representam aproximadamente 49,19%.
   * **UTI total existente**: Corresponde a cerca de 49,26%.
   * **UTI total SUS**: SP também lidera em UTIs SUS, com aproximadamente 51,04%.

O estado de São Paulo (SP) é o com maior participação tanto em leitos existentes quanto em UTIs. Minas Gerais (MG) tem uma proporção significativa de leitos SUS e UTIs SUS. E o Rio de Janeiro (RJ) se destaca na proporção de UTIs totais existentes.



**UF LEITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS**

DF 21.901803 17.867900 31.819206 22.357564

GO 44.081821 44.235556 34.986474 41.178782

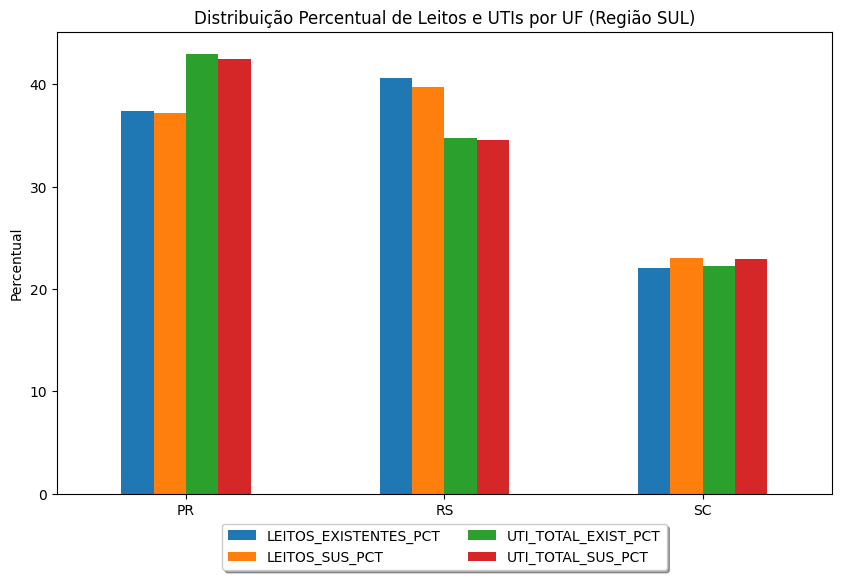
MS 14.183881 15.146796 12.680343 17.288802

MT 19.832495 22.749747 20.513977 19.174853

Com base nos dados dos percentuais de leitos e UTIs por UF na região Centro-Oeste, podemos fazer as seguintes análises:

1. **Distrito Federal (DF)**:
   * **Leitos existentes**: Representam cerca de 21,90% do total da região.
   * **Leitos SUS**: Correspondem a aproximadamente 17,87% do total.
   * **UTI total existente**: É responsável por cerca de 31,82%.
   * **UTI total SUS**: Representa aproximadamente 22,36%.
2. **Goiás (GO)**:
   * **Leitos existentes**: GO lidera em leitos existentes, com cerca de 44,08% do total da região.
   * **Leitos SUS**: Também representa aproximadamente 44,24%.
   * **UTI total existente**: Corresponde a cerca de 34,99%.
   * **UTI total SUS**: GO tem uma proporção significativa de UTIs SUS, com aproximadamente 41,18%.
3. **Mato Grosso do Sul (MS)**:
   * **Leitos existentes**: MS possui cerca de 14,18% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 15,15%.
   * **UTI total existente**: Corresponde a cerca de 12,68%.
   * **UTI total SUS**: É responsável por aproximadamente 17,29%.
4. **Mato Grosso (MT)**:
   * **Leitos existentes**: MT tem cerca de 19,83% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 22,75%.
   * **UTI total existente**: Corresponde a cerca de 20,51%.
   * **UTI total SUS**: MT também tem uma proporção significativa de UTIs SUS, com aproximadamente 19,17%.

O estado de Goiás (GO) lidera em leitos existentes e UTIs totais existentes. O Distrito Federal (DF) tem uma proporção significativa de UTIs totais existentes. Mato Grosso (MT) e Mato Grosso do Sul (MS) têm proporções semelhantes de leitos e UTIs.



**UF LEITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS**

PR 37.392405 37.165795 42.990323 42.500467

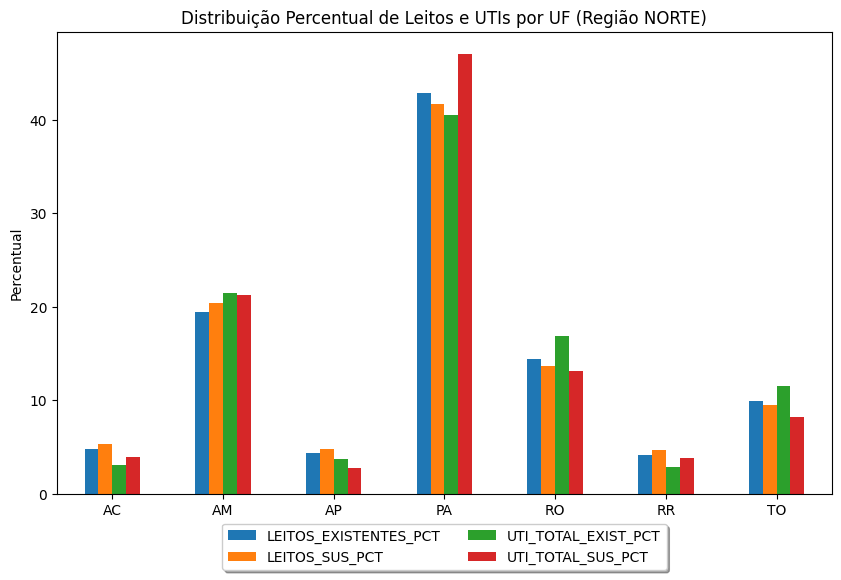
RS 40.610692 39.772236 34.765164 34.586887

SC 21.996903 23.061969 22.244513 22.912646

Com base nos dados dos percentuais de leitos e UTIs por UF na região Sul, podemos fazer as seguintes análises:

1. **Paraná (PR)**:
   * **Leitos existentes**: Representam cerca de 37,39% do total da região.
   * **Leitos SUS**: Correspondem a aproximadamente 37,17% do total.
   * **UTI total existente**: É responsável por cerca de 42,99%.
   * **UTI total SUS**: Representa aproximadamente 42,50%.
2. **Rio Grande do Sul (RS)**:
   * **Leitos existentes**: RS lidera em leitos existentes, com cerca de 40,61% do total da região.
   * **Leitos SUS**: Também representa aproximadamente 39,77%.
   * **UTI total existente**: Corresponde a cerca de 34,77%.
   * **UTI total SUS**: RS tem uma proporção significativa de UTIs SUS, com aproximadamente 34,59%.
3. **Santa Catarina (SC)**:
   * **Leitos existentes**: SC possui cerca de 21,99% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 23,06%.
   * **UTI total existente**: Corresponde a cerca de 22,24%.
   * **UTI total SUS**: É responsável por aproximadamente 22,91%.

O Rio Grande do Sul (RS) lidera em leitos existentes, enquanto Paraná (PR) tem a maior proporção de UTIs totais existentes. Já Santa Catarina (SC) tem proporções intermediárias em leitos e UTIs.



**UF LEITOS EXISTENTES LEITOS SUS UTI TOTAL EXIST UTI\_TOTAL\_SUS**

AC 4.737703 5.278708 3.036837 3.867579

AM 19.478499 20.396998 21.500449 21.247812

AP 4.396517 4.784426 3.755615 2.721630

PA 42.888584 41.697185 40.494160 47.079421

RO 14.442922 13.669146 16.873315 13.098838

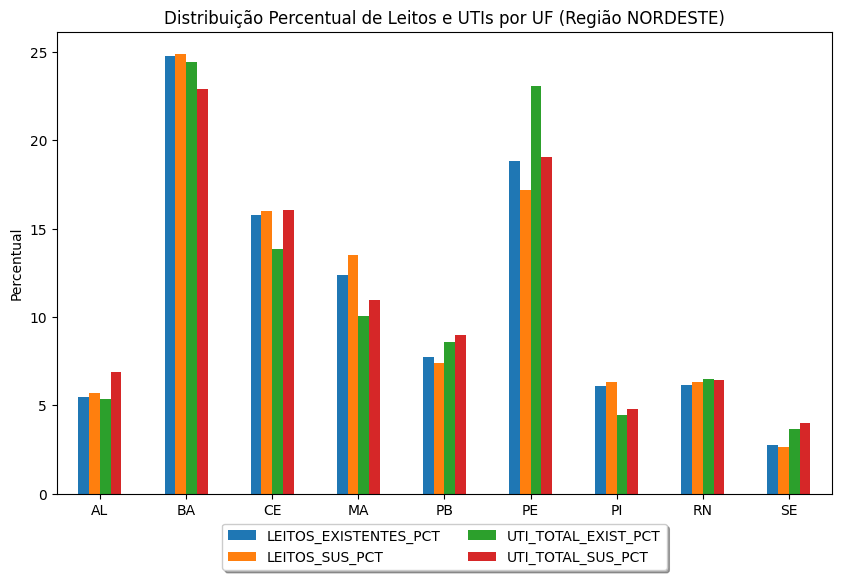
RR 4.118089 4.674336 2.803235 3.819831

TO 9.937685 9.499202 11.536388 8.164889

Com base nos dados dos percentuais de leitos e UTIs por UF na região Norte, podemos fazer as seguintes análises:

1. **Acre (AC)**:
   * **Leitos existentes**: Representam cerca de 4,74% do total da região.
   * **Leitos SUS**: Correspondem a aproximadamente 5,28% do total.
   * **UTI total existente**: É responsável por cerca de 3,04%.
   * **UTI total SUS**: Representa aproximadamente 3,87%.
2. **Amazonas (AM)**:
   * **Leitos existentes**: AM possui cerca de 19,48% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 20,40%.
   * **UTI total existente**: Corresponde a cerca de 21,50%.
   * **UTI total SUS**: AM também tem uma proporção significativa de UTIs SUS, com aproximadamente 21,25%.
3. **Amapá (AP)**:
   * **Leitos existentes**: AP tem cerca de 4,40% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 4,78%.
   * **UTI total existente**: Corresponde a cerca de 3,76%.
   * **UTI total SUS**: É responsável por aproximadamente 2,72%.
4. **Pará (PA)**:
   * **Leitos existentes**: PA lidera em leitos existentes, com cerca de 42,89% do total da região.
   * **Leitos SUS**: Também representa aproximadamente 41,70%.
   * **UTI total existente**: Corresponde a cerca de 40,49%.
   * **UTI total SUS**: PA tem a maior proporção de UTIs SUS, com aproximadamente 47,08%.
5. **Rondônia (RO)**:
   * **Leitos existentes**: RO possui cerca de 14,44% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 13,67%.
   * **UTI total existente**: Corresponde a cerca de 16,87%.
   * **UTI total SUS**: É responsável por aproximadamente 13,10%.
6. **Roraima (RR)**:
   * **Leitos existentes**: RR tem cerca de 4,12% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 4,67%.
   * **UTI total existente**: Corresponde a cerca de 2,80%.
   * **UTI total SUS**: RR também tem uma proporção significativa de UTIs SUS, com aproximadamente 3,82%.
7. **Tocantins (TO)**:
   * **Leitos existentes**: TO possui cerca de 9,94% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 9,50%.
   * **UTI total existente**: Corresponde a cerca de 11,54%.
   * **UTI total SUS**: É responsável por aproximadamente 8,16%.

O estado do Pará (PA) lidera em leitos existentes e UTIs totais existentes. Já o Amazonas (AM) e Acre (AC) têm proporções significativas de leitos e UTIs. E Roraima (RR) tem a menor proporção de UTIs totais existentes.



**UF LEITOS\_EXISTENTES LEITOS\_SUS UTI\_TOTAL\_EXIST UTI\_TOTAL\_SUS**

AL 5.467923 5.679250 5.379977 6.883356

BA 24.744443 24.903571 24.420418 22.903576

CE 15.794577 15.994384 13.826801 16.039117

MA 12.363956 13.479147 10.073812 10.941560

PB 7.749000 7.414497 8.607963 8.962064

PE 18.850406 17.202243 23.050733 19.062692

PI 6.101048 6.324223 4.475517 4.762130

RN 6.149902 6.345458 6.492359 6.420371

SE 2.778744 2.657226 3.672419 4.025133

Com base nos dados dos percentuais de leitos e UTIs por UF na região Nordeste, podemos fazer as seguintes análises:

1. **Alagoas (AL)**:
   * **Leitos existentes**: Representam cerca de 5,47% do total da região.
   * **Leitos SUS**: Correspondem a aproximadamente 5,68% do total.
   * **UTI total existente**: É responsável por cerca de 5,38%.
   * **UTI total SUS**: Representa aproximadamente 6,88%.
2. **Bahia (BA)**:
   * **Leitos existentes**: BA lidera em leitos existentes, com cerca de 24,74% do total da região.
   * **Leitos SUS**: Também representa aproximadamente 24,90%.
   * **UTI total existente**: Corresponde a cerca de 24,42%.
   * **UTI total SUS**: BA tem uma proporção significativa de UTIs SUS, com aproximadamente 22,90%.
3. **Ceará (CE)**:
   * **Leitos existentes**: CE possui cerca de 15,79% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 15,99%.
   * **UTI total existente**: Corresponde a cerca de 13,83%.
   * **UTI total SUS**: É responsável por aproximadamente 16,04%.
4. **Maranhão (MA)**:
   * **Leitos existentes**: MA tem cerca de 12,36% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 13,48%.
   * **UTI total existente**: Corresponde a cerca de 10,07%.
   * **UTI total SUS**: É responsável por aproximadamente 10,94%.
5. **Paraíba (PB)**:
   * **Leitos existentes**: PB possui cerca de 7,75% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 7,41%.
   * **UTI total existente**: Corresponde a cerca de 8,61%.
   * **UTI total SUS**: É responsável por aproximadamente 8,96%.
6. **Pernambuco (PE)**:
   * **Leitos existentes**: PE tem cerca de 18,85% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 17,20%.
   * **UTI total existente**: Corresponde a cerca de 23,05%.
   * **UTI total SUS**: PE também tem uma proporção significativa de UTIs SUS, com aproximadamente 19,06%.
7. **Piauí (PI)**:
   * **Leitos existentes**: PI possui cerca de 6,10% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 6,32%.
   * **UTI total existente**: Corresponde a cerca de 4,48%.
   * **UTI total SUS**: É responsável por aproximadamente 4,76%.
8. **Rio Grande do Norte (RN)**:
   * **Leitos existentes**: RN tem cerca de 6,15% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 6,35%.
   * **UTI total existente**: Corresponde a cerca de 6,49%.
   * **UTI total SUS**: RN também tem uma proporção significativa de UTIs SUS, com aproximadamente 6,42%.
9. **Sergipe (SE)**:
   * **Leitos existentes**: SE possui cerca de 2,78% dos leitos existentes na região.
   * **Leitos SUS**: Representam aproximadamente 2,66%.
   * **UTI total existente**: Corresponde a cerca de 3,67%.
   * **UTI total SUS**: É responsável por aproximadamente 4,03%.

O estado da **Bahia** (BA) lidera em leitos existentes, enquanto Pernambuco (PE) tem a maior proporção de UTIs totais existentes. O estado de **Pernambuco** tem uma proporção significativa de leitos existentes e leitos SUS. A porcentagem de UTIs totais existentes é alta, e a proporção de UTIs SUS também é considerável.

O estado do **Ceará** tem uma proporção significativa de leitos existentes e leitos SUS. A porcentagem de UTIs totais existentes é menor em comparação com outros estados, mas a proporção de UTIs SUS é relativamente alta.

Os estados da **Paraíba** e **Rio Grande do Norte** têm uma proporção intermediária de leitos existentes e leitos SUS. A porcentagem de UTIs totais existentes e UTIs SUS está alinhada com outros estados.

Os estados do **Maranhão** e do **Piauí** têm uma proporção menor de leitos existentes e leitos SUS. A porcentagem de UTIs totais existentes e UTIs SUS do Maranhão também é menor, mas a do estado do Piauí é relativamente baixa. Já o estado de **Sergipe** tem a menor proporção de leitos existentes e leitos SUS na região. A porcentagem de UTIs totais existentes e UTIs SUS é relativamente baixa.

Cada estado tem suas particularidades, e a distribuição dos recursos de saúde varia. Alguns estados têm foco em leitos existentes, enquanto outros priorizam UTIs.

# 

# **4. Estatística descritiva**

Apresentação da análise estatística descritiva do dataset ‘**Leitos2024**’, utilizando visualizações gráficas para explorar a distribuição e correlação dos dados. As visualizações incluem Box plot, Histograma, Scatter plot e Mapa de calor para correlação (heatmap).

**Metodologia:** A análise foi conduzida utilizando a biblioteca seaborn para visualizações estatísticas e a biblioteca matplotlib para a exibição dos gráficos. Foram criados subconjuntos de dados para focar em variáveis específicas e avaliar suas relações.

**Resultados:**

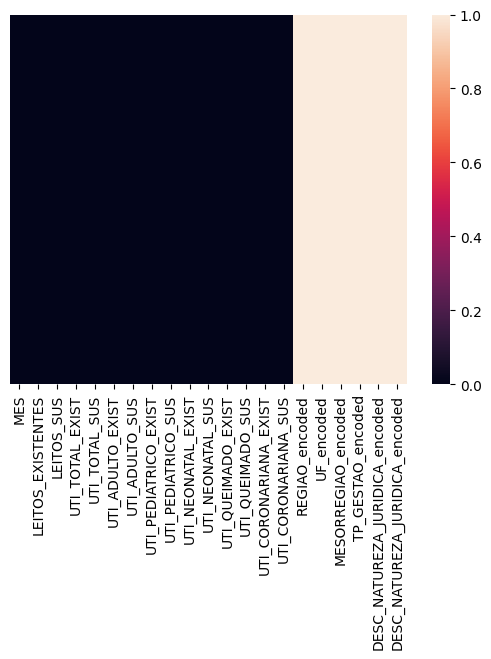
* **Mapa de Calor de Valores Nulos:** O primeiro mapa de calor indicou a ausência de valores nulos no dataset, essencial para a limpeza e preparação dos dados.
* **Correlação entre Variáveis**: Matrizes de correlação foram calculadas para subconjuntos de dados, incluindo Leitos Existentes e Leitos SUS, e diversas categorias de UTIs existentes e SUS. Mapas de calor visualizaram a força e direção das relações entre as variáveis.
* **Descrição Estatística**: A função describe forneceu estatísticas descritivas do dataset, incluindo contagem, média, desvio padrão, mínimo, quartis e máximo para cada variável numérica.
* **Box Plot**: A análise do Box Plot revelou a mediana, quartis e outliers para os leitos existentes, oferecendo insights sobre a simetria da distribuição e comparações entre categorias de ‘LEITOS\_SUS’.
* **Histograma com Gráfico Marginal Tipo ‘Rug’**: Esta visualização combinada mostrou a distribuição de frequência dos dados e detalhes granulares de cada ponto de dado.
* **Scatter Plots**: Os gráficos de dispersão gerados pelo sns.pairplot exploraram as relações entre ‘LEITOS\_EXISTENTES’ e ‘LEITOS\_SUS’, destacando correlações, tendências e distribuições dos dados.

**Código do gráfico:**

#Importar a lib seaborn para observações estatísticas

import seaborn as sns

sns.heatmap(leitos.isnull(), yticklabels = False) # cria um grafico tipo mapa de calor



* **Correlação entre Variáveis:** Foram calculadas as matrizes de correlação para três subconjuntos de dados:
  1. Leitos Existentes e Leitos SUS.
  2. Diversas categorias de UTIs existentes.
  3. Diversas categorias de UTIs SUS.
* Cada matriz de correlação foi visualizada através de mapas de calor, destacando a força e direção das relações entre as variáveis.

**Código da análise:**

# cria um subconjunto contendo apenas as colunas especificadas

subset1 = leitos[['LEITOS\_EXISTENTES', 'LEITOS\_SUS']] #criando um subconjunto

subset2 = leitos[['UTI\_TOTAL\_EXIST', 'UTI\_ADULTO\_EXIST', 'UTI\_PEDIATRICO\_EXIST', 'UTI\_NEONATAL\_EXIST', 'UTI\_QUEIMADO\_EXIST', 'UTI\_CORONARIANA\_EXIST']] #criando um subconjunto

subset3 = leitos[['UTI\_TOTAL\_SUS', 'UTI\_ADULTO\_SUS', 'UTI\_PEDIATRICO\_SUS', 'UTI\_NEONATAL\_SUS', 'UTI\_QUEIMADO\_SUS', 'UTI\_CORONARIANA\_SUS']] #criando um subconjunto

matriz\_rel1 = subset1.corr() #definindo a matriz de correlação

matriz\_rel2 = subset2.corr() #definindo a matriz de correlação

matriz\_rel3 = subset3.corr() #definindo a matriz de correlação

Código do gráfico:

sns.heatmap(matriz\_rel1, annot = True, cmap = 'coolwarm') #coolwarm é útil para definir cores em mapas de calor (azul e vermelho) #annot serve para fazer dados importantes/informações serem expostas no gráfico

plt.title('Mapa de calor - Leitos Existentes x Leitos SUS')

plt.show()

sns.heatmap(matriz\_rel2, annot = True, cmap = 'coolwarm') #coolwarm é útil para definir cores em mapas de calor (azul e vermelho) #annot serve para fazer dados importantes/informações serem expostas no gráfico

plt.title('Mapa de calor - UTI EXISTENTES')

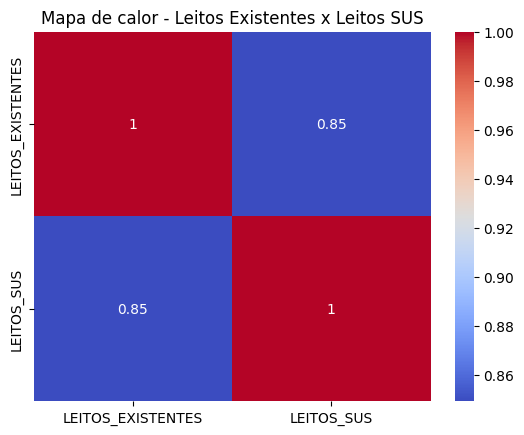
plt.show()

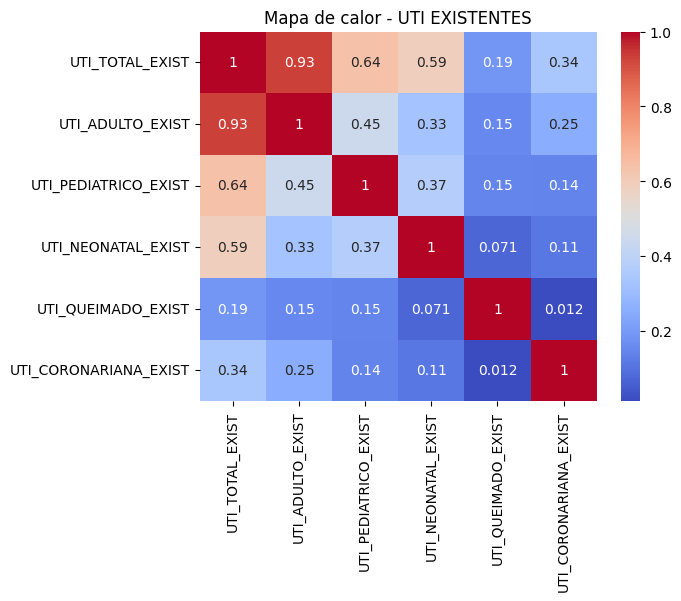
sns.heatmap(matriz\_rel3, annot = True, cmap = 'coolwarm') #coolwarm é útil para definir cores em mapas de calor (azul e vermelho) #annot serve para fazer dados importantes/informações serem expostas no gráfico

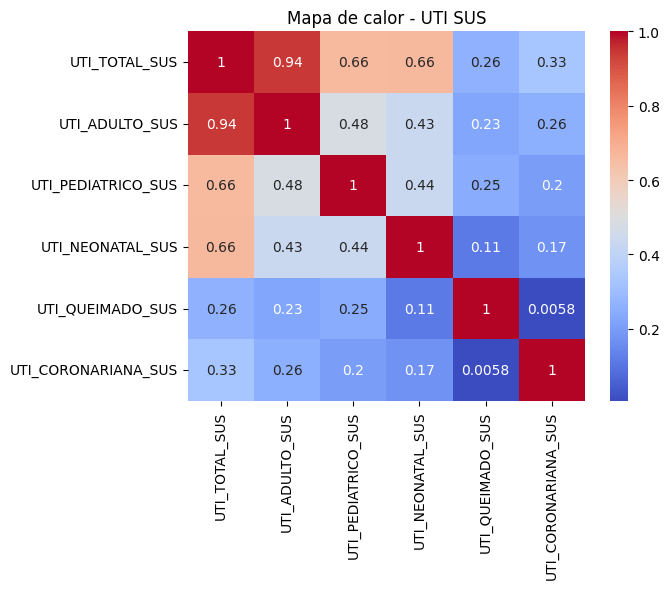
plt.title('Mapa de calor - UTI SUS')

plt.show()

sns.heatmap(leitos.isnull(), yticklabels = False, cbar=False)



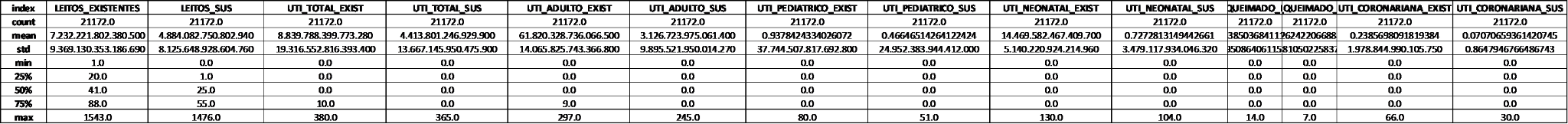


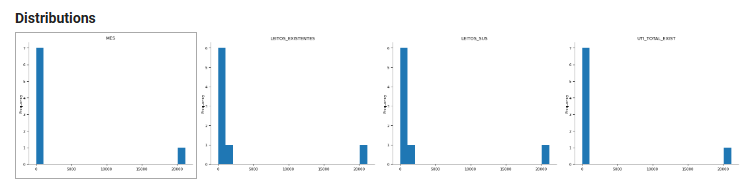


* **Descrição Estatística:** A função describe foi utilizada para fornecer estatísticas descritivas do dataset, incluindo contagem, média, desvio padrão, mínimo, quartis e máximo para cada variável numérica.

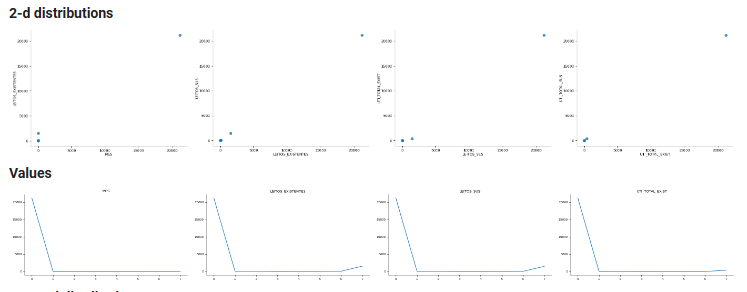
**Código da analise:**

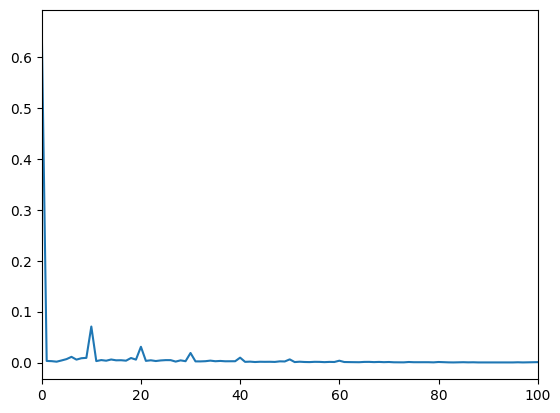
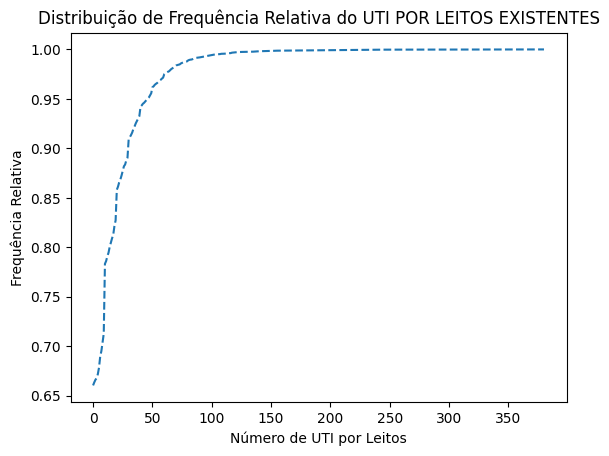
leitos.describe()

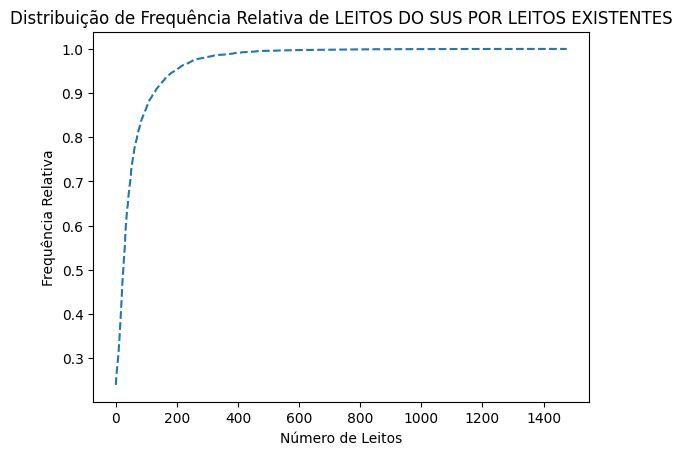
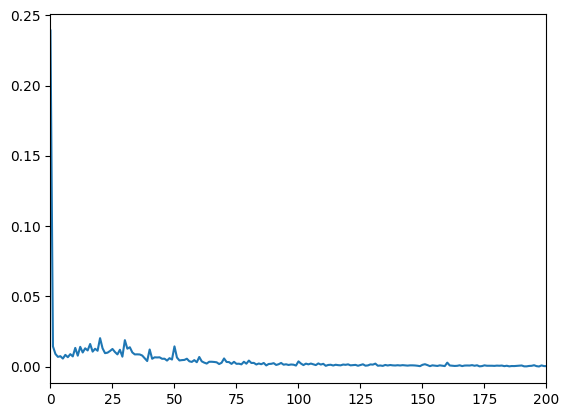


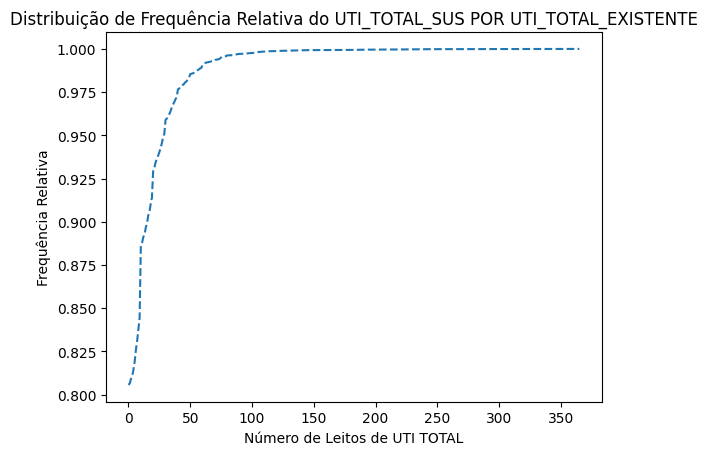
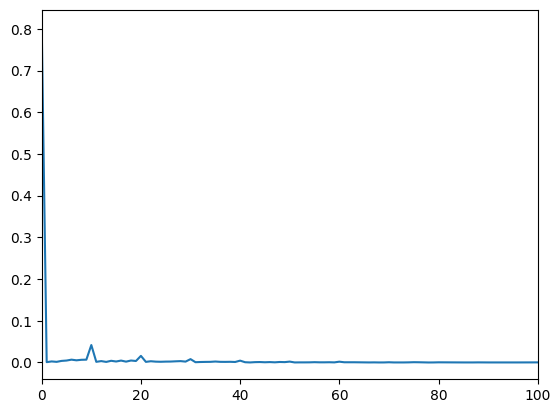


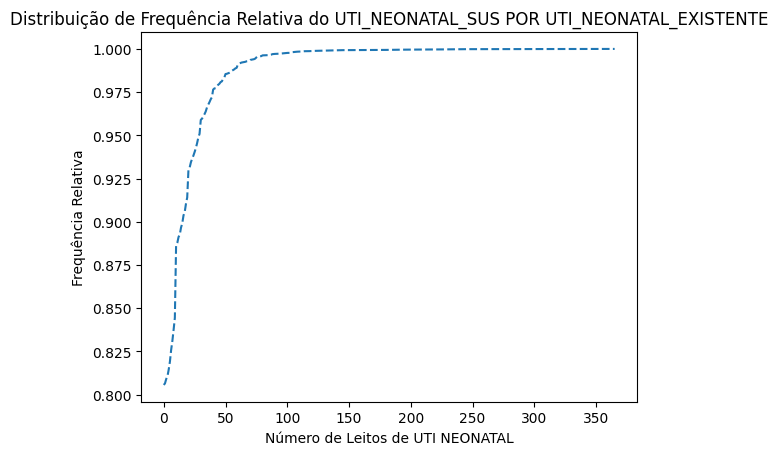
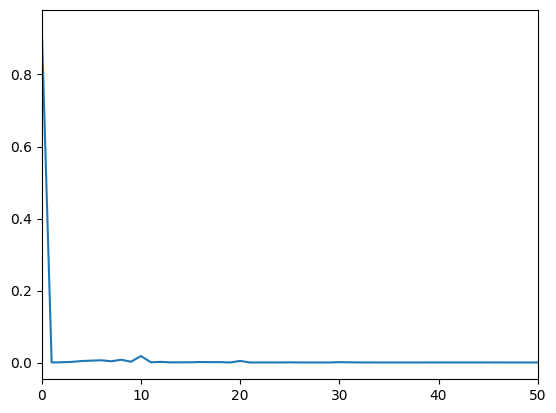


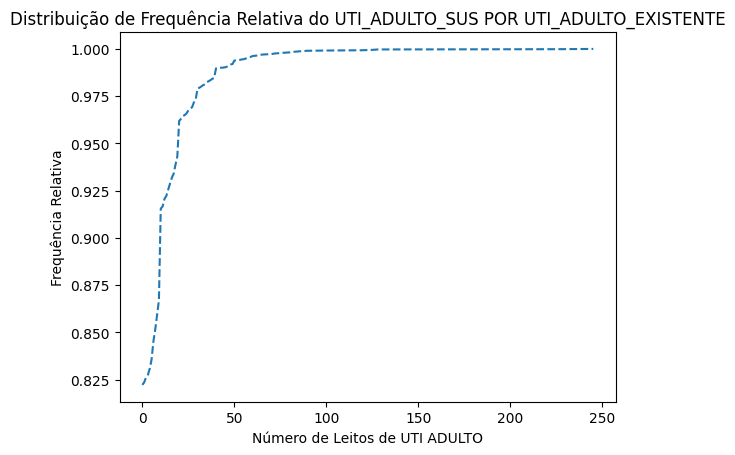
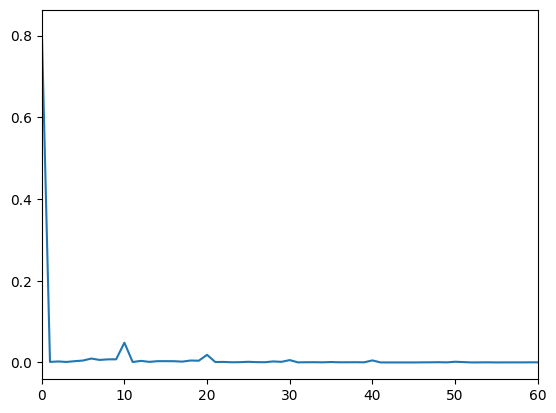


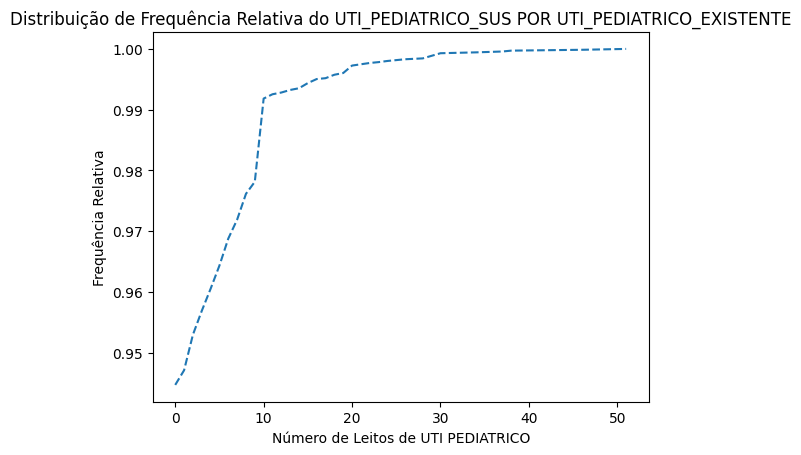
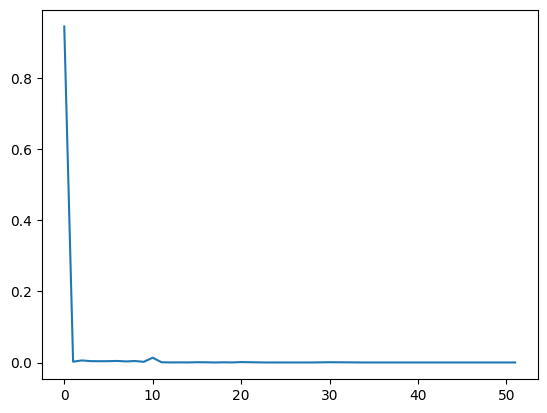
 

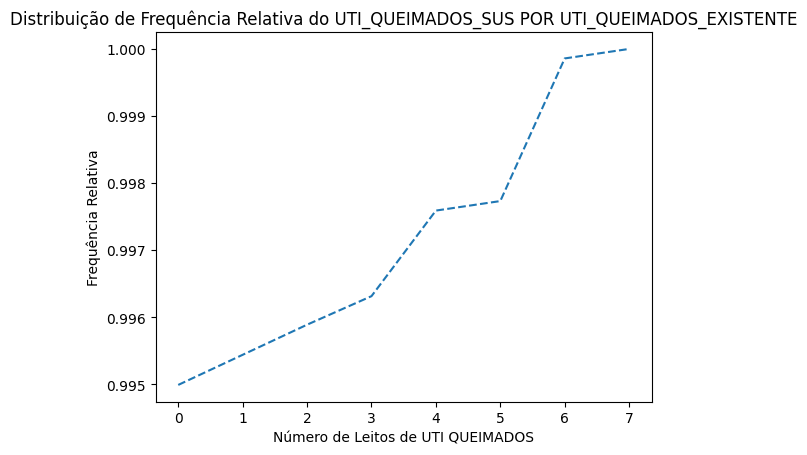
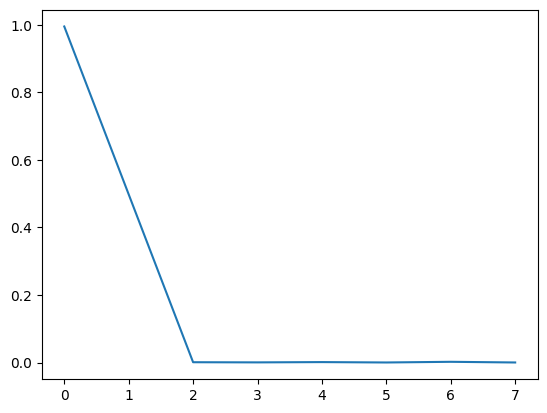


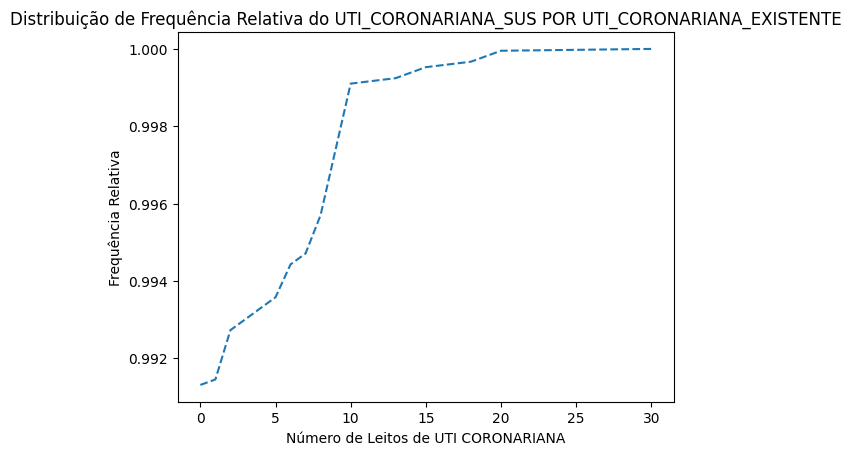
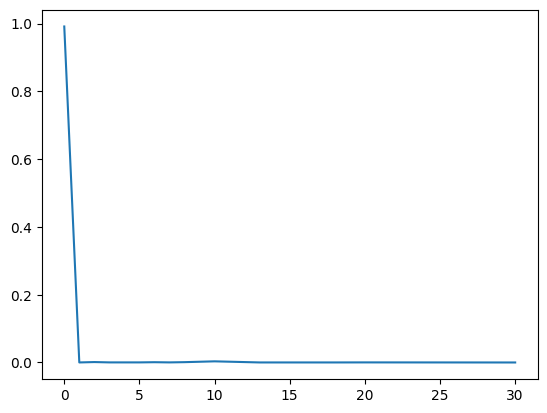




****

****

****

****

**Código dos gráficos:**

import matplotlib.pyplot as plt #biblioteca que traz diversos gráficos

plt.plot(fp) #comando plot serve para criar gráficos de linhas

grafico = fp.cumsum()

plt.plot(grafico, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI POR LEITOS EXISTENTES')

plt.xlabel('Número de UTI por Leitos')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp1) #comando plot serve para criar gráficos de linhas

grafico1 = fp1.cumsum()

plt.plot(grafico1, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa de LEITOS DO SUS POR LEITOS EXISTENTES')

plt.xlabel('Número de Leitos')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp2) #comando plot serve para criar gráficos de linhas

grafico2 = fp2.cumsum()

plt.plot(grafico2, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI\_TOTAL\_SUS POR UTI\_TOTAL\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI TOTAL')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp3) #comando plot serve para criar gráficos de linhas

grafico3 = fp3.cumsum()

plt.plot(grafico2, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI\_NEONATAL\_SUS POR UTI\_NEONATAL\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI NEONATAL')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp4) #comando plot serve para criar gráficos de linhas

grafico4 = fp4.cumsum()

plt.plot(grafico4, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI\_ADULTO\_SUS POR UTI\_ADULTO\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI ADULTO')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp5) #comando plot serve para criar gráficos de linhas

grafico5 = fp5.cumsum()

plt.plot(grafico5, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI\_PEDIATRICO\_SUS POR UTI\_PEDIATRICO\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI PEDIATRICO')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp6) #comando plot serve para criar gráficos de linhas

grafico6 = fp6.cumsum()

plt.plot(grafico6, linestyle='--') #definindo o estilo da linha como tracejada

plt.title('Distribuição de Frequência Relativa do UTI\_QUEIMADOS\_SUS POR UTI\_QUEIMADOS\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI QUEIMADOS')

plt.ylabel('Frequência Relativa')

plt.show()

plt.plot(fp7) #comando plot serve para criar gráficos de linhas

grafico7 = fp7.cumsum()

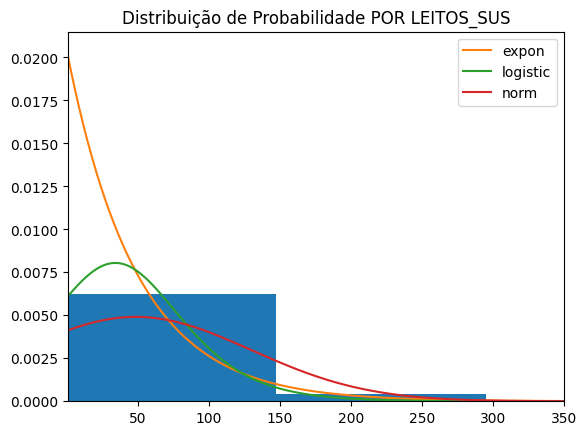
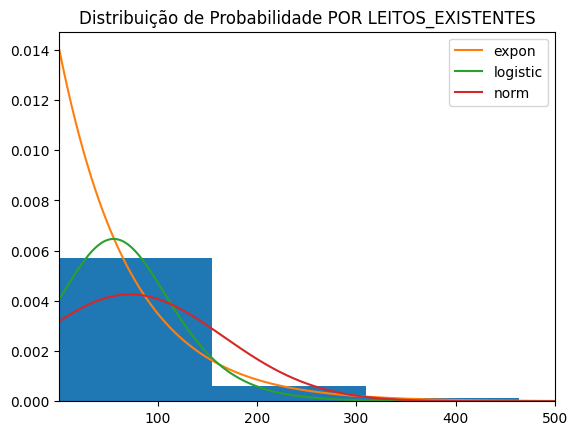
plt.plot(grafico7, linestyle='--') #definindo o estilo da linha como tracejada

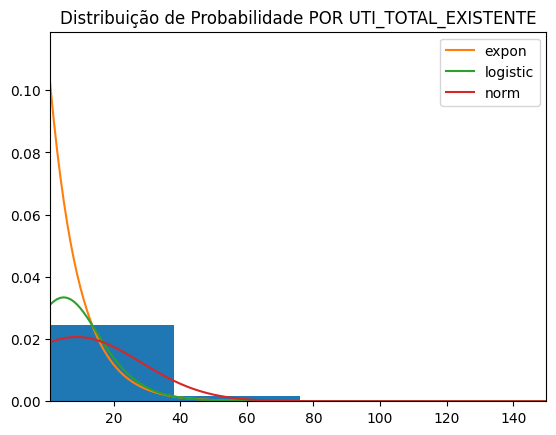
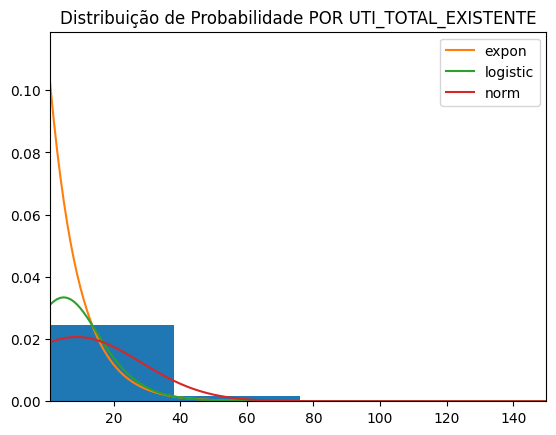
plt.title('Distribuição de Frequência Relativa do UTI\_CORONARIANA\_SUS POR UTI\_CORONARIANA\_EXISTENTE')

plt.xlabel('Número de Leitos de UTI CORONARIANA')

plt.ylabel('Frequência Relativa')

plt.show()





**Código dos gráficos:**

import scipy #bibloteca para resolução de problemas de matemática e engenharia

import scipy.stats #módulo com grande variadade de funções

size=700000 #tamanho da escala dos eixos coordenados

plt.xlim(1, 450)

x = np.arange(size) #a função arange retorna uma matriz de valores igualmente espaçados dentro do intervalo especificado

y = leitos['LEITOS\_SUS'].values

h = plt.hist(y, density=True) #chamando a função histograma, o histograma aqui é unidimensional em y e density = True normaliza os dados

plt.title('Distribuição de Probabilidade POR LEITOS\_SUS')

dist\_names = ['expon', 'logistic', 'norm']

for dist\_name in dist\_names: #o comando for \_ in cria um laço

dist = getattr(scipy.stats, dist\_name) #a função getattr retorna o valor de um atributo de um objeto

params = dist.fit(y)

arg = params[:-2] #cria uma nova lista mostrando os primeiros elementos excluindo os 2 últimos elementos

loc = params[-2] #localiza o penúltimo elemento

scale = params[-1] #localiza o último elemento

if arg:

pdf\_fitted = dist.pdf(x, \*arg, loc=loc, scale=scale) #\*arg permite que a função aceite números variáveis de argumentos posicionais; loc = loc é a mesma coisa de loc = loc + loc

else:

pdf\_fitted = dist.pdf(x, loc=loc, scale=scale)

plt.plot(pdf\_fitted, label=dist\_name)

plt.legend(loc= 'upper right')

size=700000 #tamanho da escala dos eixos coordenados

plt.xlim(1, 350)

x = np.arange(size) #a função arange retorna uma matriz de valores igualmente espaçados dentro do intervalo especificado

y = leitos['LEITOS\_EXISTENTES'].values

h = plt.hist(y, density=True) #chamando a função histograma, o histograma aqui é unidimensional em y e density = True normaliza os dados

plt.title('Distribuição de Probabilidade POR LEITOS\_EXISTENTE')

dist\_names = ['expon', 'logistic', 'norm']

for dist\_name in dist\_names: #o comando for \_ in cria um laço

dist = getattr(scipy.stats, dist\_name) #a função getattr retorna o valor de um atributo de um objeto

params = dist.fit(y)

arg = params[:-2] #cria uma nova lista mostrando os primeiros elementos excluindo os 2 últimos elementos

loc = params[-2] #localiza o penúltimo elemento

scale = params[-1] #localiza o último elemento

if arg:

pdf\_fitted = dist.pdf(x, \*arg, loc=loc, scale=scale) #\*arg permite que a função aceite números variáveis de argumentos posicionais; loc = loc é a mesma coisa de loc = loc + loc

else:

pdf\_fitted = dist.pdf(x, loc=loc, scale=scale)

plt.plot(pdf\_fitted, label=dist\_name)

plt.legend(loc= 'upper right')

size=700000 #tamanho da escala dos eixos coordenados

plt.xlim(1, 150)

x = np.arange(size) #a função arange retorna uma matriz de valores igualmente espaçados dentro do intervalo especificado

y = leitos['UTI\_TOTAL\_SUS'].values

h = plt.hist(y, density=True) #chamando a função histograma, o histograma aqui é unidimensional em y e density = True normaliza os dados

plt.title('Distribuição de Probabilidade POR UTI\_TOTAL\_EXIST')

dist\_names = ['expon', 'logistic', 'norm']

for dist\_name in dist\_names: #o comando for \_ in cria um laço

dist = getattr(scipy.stats, dist\_name) #a função getattr retorna o valor de um atributo de um objeto

params = dist.fit(y)

arg = params[:-2] #cria uma nova lista mostrando os primeiros elementos excluindo os 2 últimos elementos

loc = params[-2] #localiza o penúltimo elemento

scale = params[-1] #localiza o último elemento

if arg:

pdf\_fitted = dist.pdf(x, \*arg, loc=loc, scale=scale) #\*arg permite que a função aceite números variáveis de argumentos posicionais; loc = loc é a mesma coisa de loc = loc + loc

else:

pdf\_fitted = dist.pdf(x, loc=loc, scale=scale)

plt.plot(pdf\_fitted, label=dist\_name)

plt.legend(loc= 'upper right')

import scipy #bibloteca para resolução de problemas de matemática e engenharia

import scipy.stats #módulo com grande variadade de funções

size=700000 #tamanho da escala dos eixos coordenados

plt.xlim(1, 150)

x = np.arange(size) #a função arange retorna uma matriz de valores igualmente espaçados dentro do intervalo especificado

y = leitos['UTI\_TOTAL\_SUS'].values

h = plt.hist(y, density=True) #chamando a função histograma, o histograma aqui é unidimensional em y e density = True normaliza os dados

plt.title('Distribuição de Probabilidade POR UTI\_TOTAL\_SUS')

dist\_names = ['expon', 'logistic', 'norm']

for dist\_name in dist\_names: #o comando for \_ in cria um laço

dist = getattr(scipy.stats, dist\_name) #a função getattr retorna o valor de um atributo de um objeto

params = dist.fit(y)

arg = params[:-2] #cria uma nova lista mostrando os primeiros elementos excluindo os 2 últimos elementos

loc = params[-2] #localiza o penúltimo elemento

scale = params[-1] #localiza o último elemento

if arg:

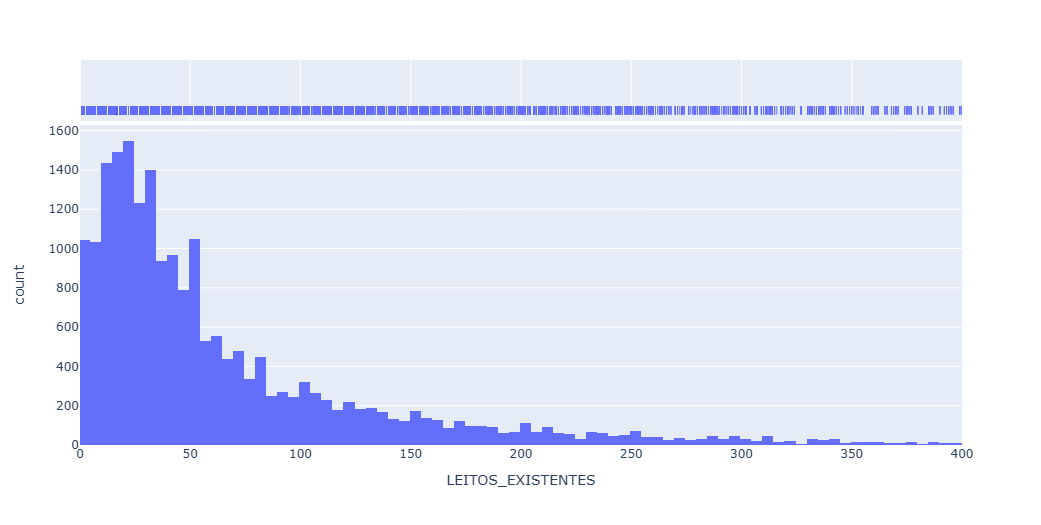
pdf\_fitted = dist.pdf(x, \*arg, loc=loc, scale=scale) #\*arg permite que a função aceite números variáveis de argumentos posicionais; loc = loc é a mesma coisa de loc = loc + loc

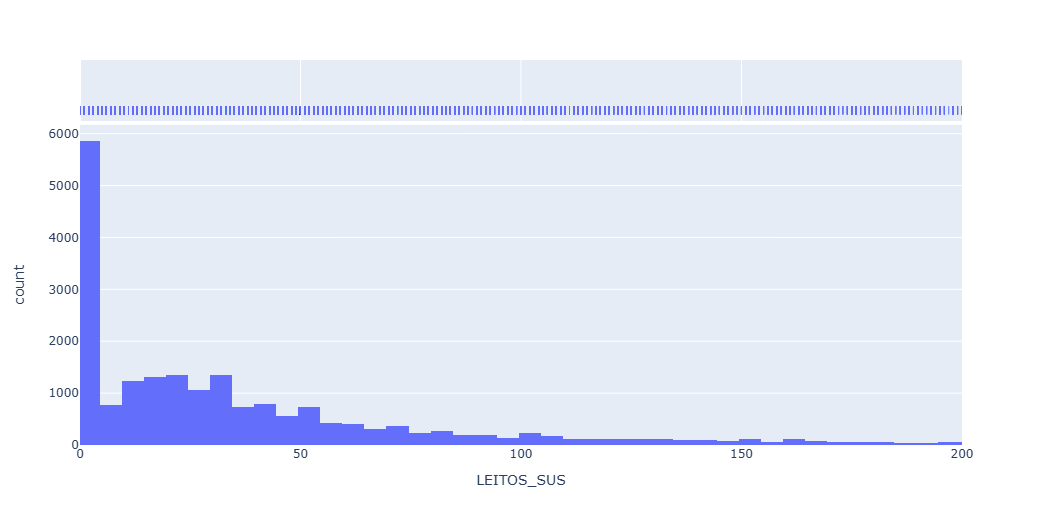
else:

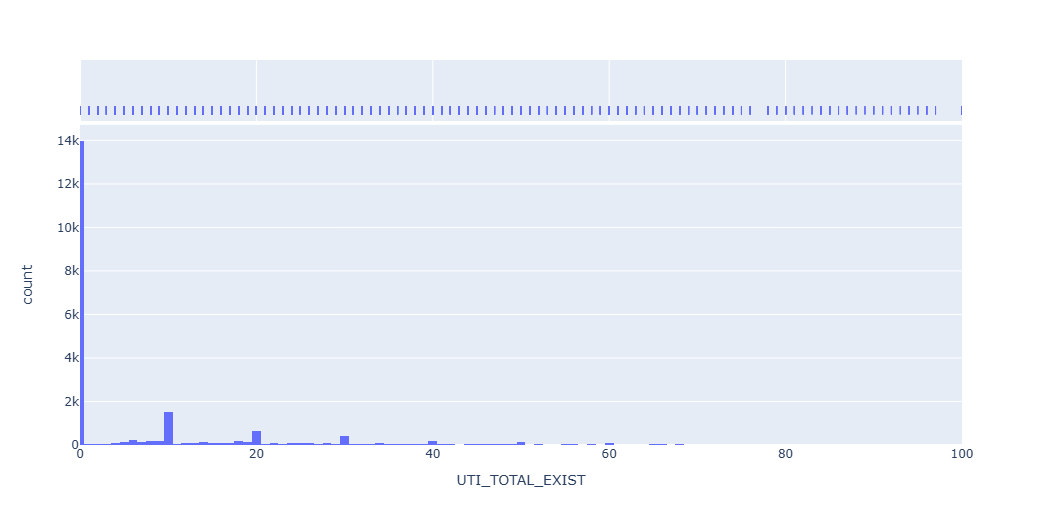
pdf\_fitted = dist.pdf(x, loc=loc, scale=scale)

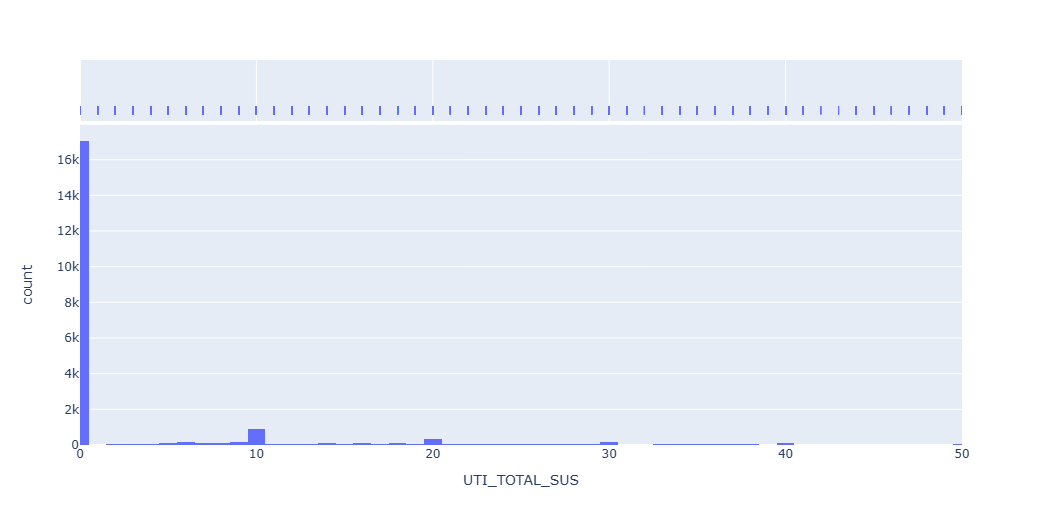
plt.plot(pdf\_fitted, label=dist\_name)

plt.legend(loc= 'upper right')









**Código dos gráficos:**

import plotly.express as px #plotly é uma biblioteca com gráficos exclusivos para visualização de dados

figA = px.histogram(leitos, x='LEITOS\_EXISTENTES', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e ocomando rug cria "marcas"

figA.update\_xaxes(range=[0, 400]) # Definindo os limites do eixo X

figA.show()

figB = px.histogram(leitos, x='LEITOS\_SUS', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e ocomando rug cria "marcas"

figB.update\_xaxes(range=[0, 200]) # Definindo os limites do eixo X

figB.show()

figC = px.histogram(leitos, x='UTI\_TOTAL\_EXIST', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e ocomando rug cria "marcas"

figC.update\_xaxes(range=[0, 100]) # Definindo os limites do eixo X

figC.show()

figD = px.histogram(leitos, x='UTI\_TOTAL\_SUS', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e ocomando rug cria "marcas"

figD.update\_xaxes(range=[0, 50]) # Definindo os limites do eixo X

figD.show()

O histograma complementado por um gráfico marginal tipo ‘rug’ forneceu uma representação detalhada da distribuição dos leitos existentes. O histograma principal ilustrou a frequência dos dados, enquanto o gráfico marginal ‘rug’ destacou a localização exata de cada ponto de dado. Através desta combinação, podemos observar:

* **Distribuição de Frequência**: O histograma mostra como os dados de ‘LEITOS\_EXISTENTES’,‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’ e ‘UTI\_TOTAL\_SUS’ estão distribuídos em termos de frequência, permitindo identificar padrões como picos e assimetrias na distribuição.
* **Detalhes dos Dados Individuais**: O gráfico marginal tipo ‘rug’ adicionou uma linha fina para cada ponto de dado ao longo do eixo x, oferecendo uma visão granular da distribuição dos dados.
* **Densidade dos Dados**: A proximidade das marcas no gráfico ‘rug’ indica áreas de maior densidade, onde há uma concentração maior de leitos existentes.
* **Identificação de Clusters**: Agrupamentos de marcas no gráfico ‘rug’ podem sugerir a formação de clusters, indicando intervalos com alta concentração de valores.
* **Complemento ao Histograma**: Enquanto o histograma forneceu uma visão geral, o gráfico ‘rug’ oferece um nível de detalhe que ajuda a identificar onde os dados estão mais concentrados.

Essa análise de visual duplo é particularmente útil para entender não apenas a tendência central e a dispersão dos dados, mas também para reconhecer a distribuição exata dos leitos existentes. Isso pode ser crucial para identificar necessidades específicas de alocação de recursos e para planejamento estratégico no sistema de saúde.

**Código dos gráficos:**

import plotly.express as px #plotly é uma biblioteca com gráficos exclusivos para visualização de dados

figA = px.histogram(leitos, x='LEITOS\_EXISTENTES', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e o comando rug cria "marcas"

figA.show()

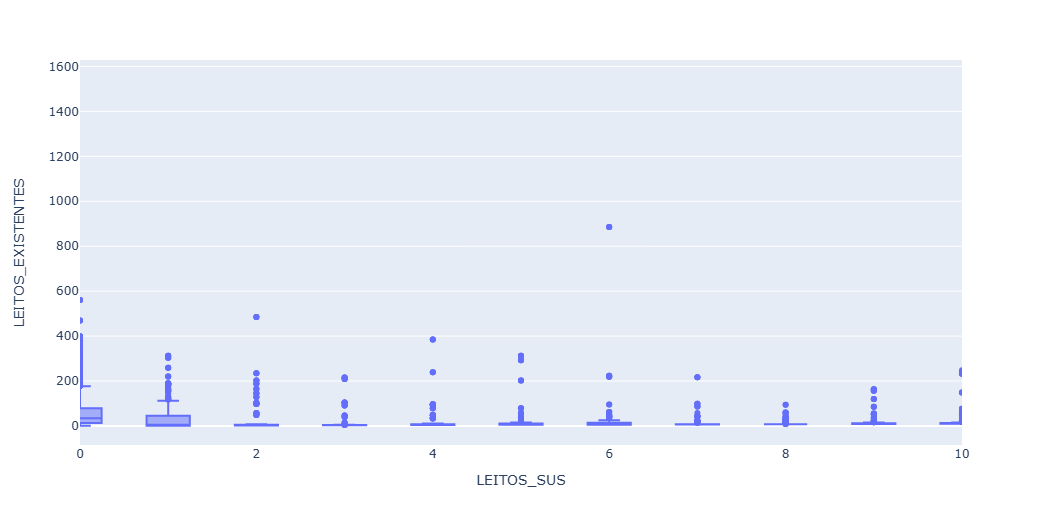
figB = px.histogram(leitos, x='LEITOS\_SUS', marginal = 'rug')

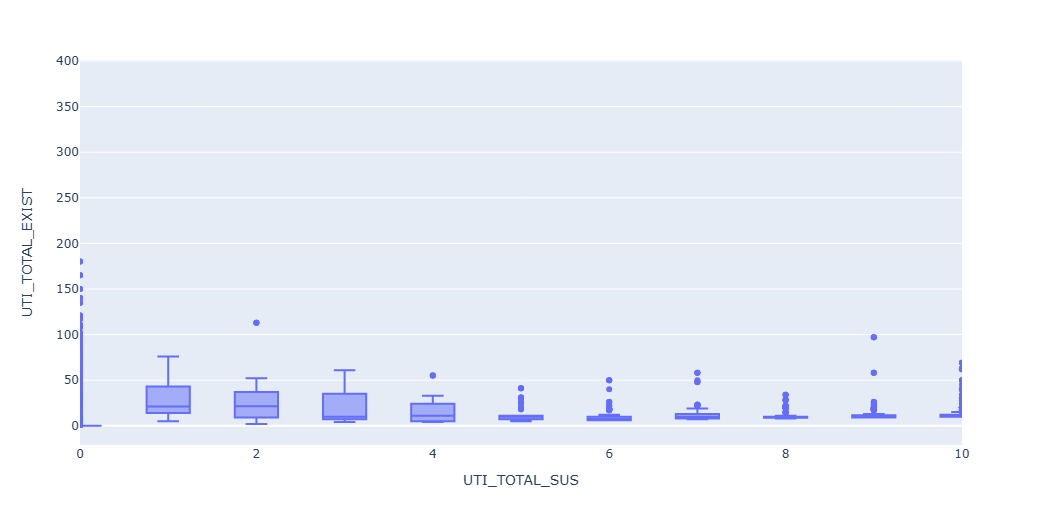
figB.show()

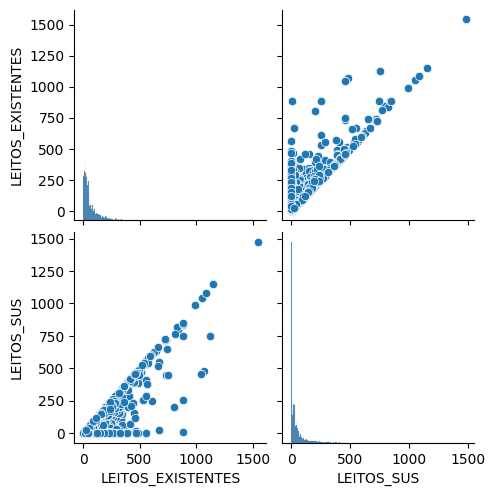
figC = px.histogram(leitos, x='UTI\_TOTAL\_EXIST', marginal = 'rug')

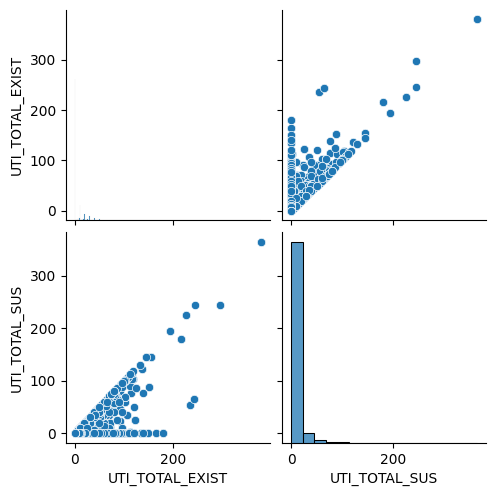
figC.show()

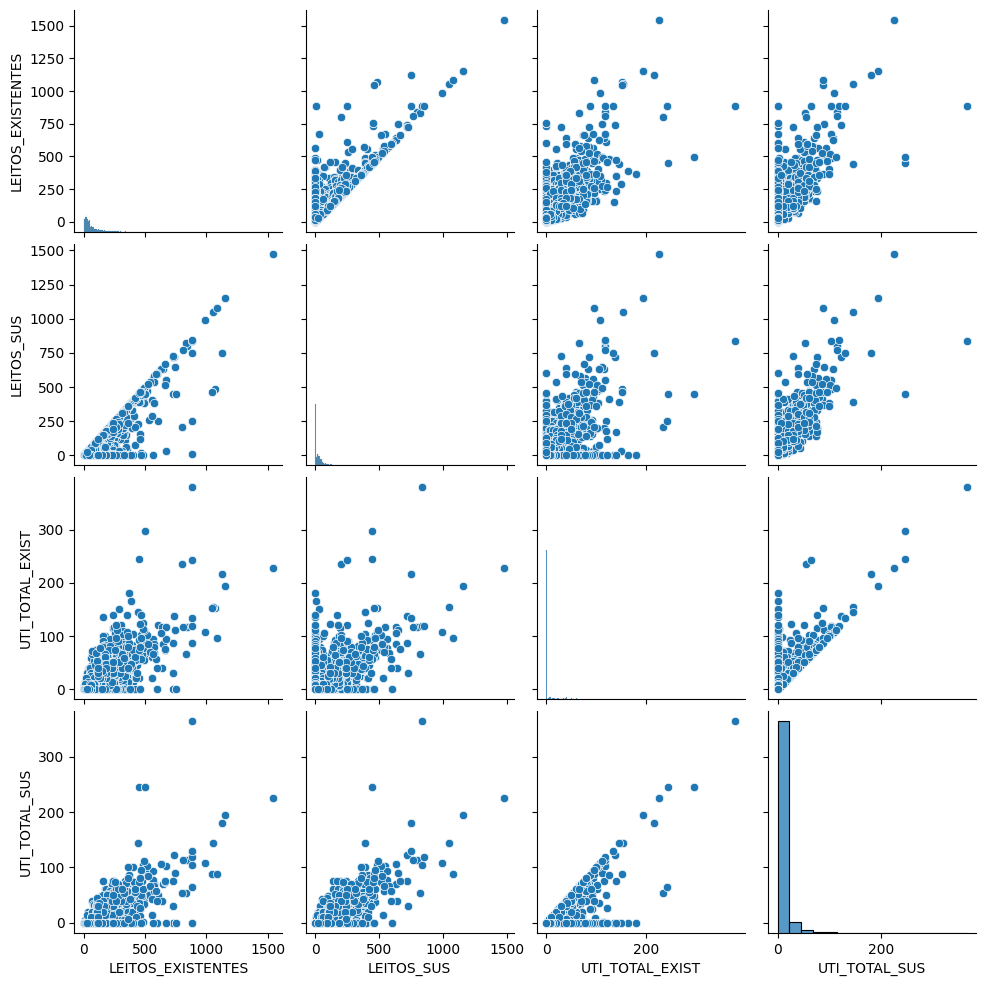
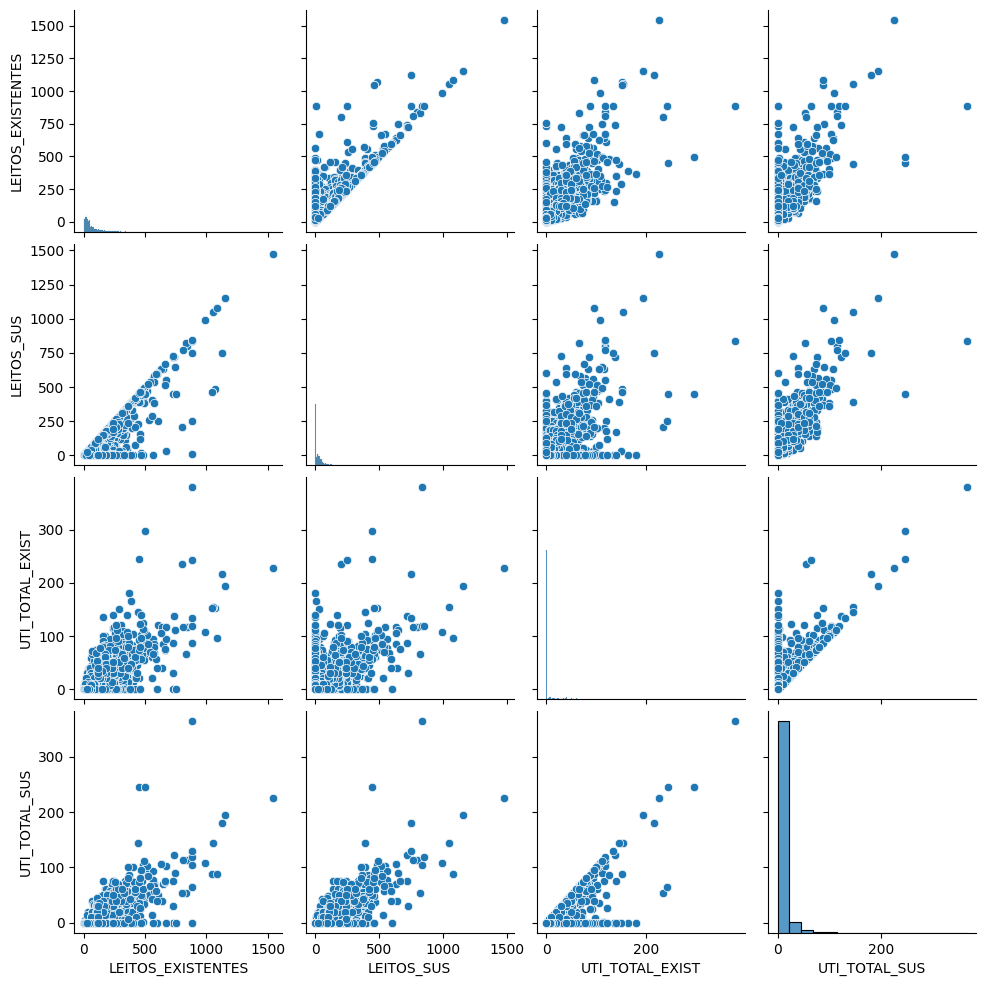
figD = px.histogram(leitos, x='UTI\_TOTAL\_SUS', marginal = 'rug') figD.show()











**Código dos gráficos:**

import seaborn as sns

col = ['LEITOS\_EXISTENTES', 'LEITOS\_SUS']

sns.pairplot(leitos[col]) #cria pares de gráficos

col2 = ['UTI\_TOTAL\_EXIST', 'UTI\_TOTAL\_SUS']

sns.pairplot(leitos[col2]) #cria pares de gráficos

col3 = ['LEITOS\_EXISTENTES', 'LEITOS\_SUS', 'UTI\_TOTAL\_EXIST', 'UTI\_TOTAL\_SUS']

sns.pairplot(leitos[col3])

Os gráficos mostram uma visualização da distribuição dos leitos existentes segmentados pelos leitos disponibilizados pelo SUS. Através deste gráfico, pode-se observar as seguintes características:

* **Mediana**: A linha central da caixa indica a mediana dos leitos existentes, que representa o valor médio do conjunto de dados, dividindo-o em duas partes iguais.
* **Quartis**: As bordas da caixa representam o primeiro e terceiro quartis, mostrando a distribuição intermediária dos dados. A distância entre esses quartis, conhecida como intervalo interquartil, fornece uma medida da dispersão e indica onde se concentra a maior parte dos dados.
* **Outliers**: Pontos que aparecem fora dos ‘bigodes’ do gráfico são considerados outliers. Estes representam hospitais que têm um número de leitos existentes significativamente diferente da norma, seja muito acima ou abaixo dos outros estabelecimentos.
* **Simetria**: A posição da mediana dentro da caixa pode nos informar sobre a simetria da distribuição. Uma mediana mais próxima do quartil superior ou inferior sugere assimetria na distribuição dos leitos existentes.
* **Comparação entre Categorias**: Se o gráfico for dividido por categorias de ‘LEITOS\_SUS’, podemos comparar a distribuição de leitos existentes entre essas categorias, o que pode indicar diferenças regionais ou por unidade federativa.

A análise do Box Plot é fundamental para identificar padrões e possíveis áreas de melhoria na alocação de leitos e recursos do SUS. Os insights obtidos podem auxiliar na tomada de decisões estratégicas para otimizar a distribuição de leitos e melhorar o acesso à saúde.

**Análise do código:**

import plotly.express as px #plotly é uma biblioteca com gráficos exclusivos para visualização de dados

fig = px.histogram(leitos, x='LEITOS\_EXISTENTES', marginal = 'rug') #função histograma e o comando marginal cria histogramas marginais e ocomando rug cria "marcas"

fig.show()

As visualizações gráficas forneceram insights sobre a estrutura dos dados. Os gráficos de dispersão e mapas de calor de correlação destacaram as relações entre diferentes tipos de leitos e UTIs.

Esta análise descritiva revelou padrões importantes e relações entre as variáveis do dataset ‘Leitos2024’. As visualizações gráficas facilitaram a interpretação dos dados e podem servir como base para análises mais profundas.

A saída da função describe forneceu um resumo estatístico que pode ser usado para entender melhor a distribuição dos dados e identificar possíveis outliers ou erros de entrada.

# **5. Normal Padrão**

Apresentação da análise da variável ‘LEITOS\_EXISTENTES’,‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’ e ‘UTI\_TOTAL\_SUS’ do dataset ‘Leitos2024’, focando na aplicação de técnicas de normalização e na avaliação da distribuição normal dos dados.

**Metodologia:** A análise foi realizada em duas etapas principais:

1. **Avaliação da Distribuição Normal:**
   * Foi calculada a média (mu) e o desvio padrão (sigma) dos dados das colunas ‘LEITOS\_EXISTENTES’, ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’ e ‘UTI\_TOTAL\_SUS’.
   * Utilizou-se a função np.linspace para gerar valores no intervalo de interesse, cobrindo a maioria dos dados.
   * A função densidade de probabilidade da distribuição normal foi calculada e plotada para visualizar a distribuição dos dados.
2. **Normalização dos Dados:**
   * Empregou-se a classe StandardScaler da biblioteca sklearn.preprocessing para normalizar os dados.
   * Ajustou-se o scaler aos dados, determinando a média e o desvio padrão, e aplicou-se a transformação para obter a normalização padrão (z-score).

**Resultados:**

* A curva normal foi plotada, indicando a distribuição dos dados de ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’, ‘UTI\_TOTAL\_SUS’.

**Código do gráfico:**

# Média e desvio padrão

mu1 = np.mean(leitos['LEITOS\_EXISTENTES'])

sigma1 = np.std(leitos['LEITOS\_EXISTENTES'])

# Gerar valores para o eixo x que cubram a maioria dos dados

x = np.linspace(mu1 - 4\*sigma1, mu1 + 4\*sigma1, 1000)

# Calcular a função densidade de probabilidade da distribuição normal

y = (1 / (sigma1 \* np.sqrt(2 \* np.pi))) \* np.exp(-(x - mu1)\*\*2 / (2 \* sigma1\*\*2))

# Plotando a curva normal

plt.plot(x, y, label="Curva Normal")

plt.xlabel("Valores")

plt.ylabel("Densidade de Probabilidade")

plt.title("Curva da Distribuição Normal para LEITOS\_EXISTENTES")

plt.legend()

plt.show()

# Média e desvio padrão

mu2 = np.mean(leitos['LEITOS\_SUS'])

sigma2 = np.std(leitos['LEITOS\_SUS'])

# Gerar valores para o eixo x que cubram a maioria dos dados

x2 = np.linspace(mu2 - 4\*sigma2, mu2 + 4\*sigma2, 1000)

# Calcular a função densidade de probabilidade da distribuição normal

y = (1 / (sigma2 \* np.sqrt(2 \* np.pi))) \* np.exp(-(x - mu2)\*\*2 / (2 \* sigma2\*\*2))

# Plotando a curva normal

plt.plot(x, y, label="Curva Normal")

plt.xlabel("Valores")

plt.ylabel("Densidade de Probabilidade")

plt.title("Curva da Distribuição Normal para LEITOS\_SUS")

plt.legend()

plt.show()

# Média e desvio padrão

mu3 = np.mean(leitos['UTI\_TOTAL\_EXIST'])

sigma3 = np.std(leitos['UTI\_TOTAL\_EXIST'])

# Gerar valores para o eixo x que cubram a maioria dos dados

x = np.linspace(mu3 - 4\*sigma3, mu3 + 4\*sigma3, 1000)

# Calcular a função densidade de probabilidade da distribuição normal

y = (1 / (sigma3 \* np.sqrt(2 \* np.pi))) \* np.exp(-(x - mu3)\*\*2 / (2 \* sigma3\*\*2))

# Plotando a curva normal

plt.plot(x, y, label="Curva Normal")

plt.xlabel("Valores")

plt.ylabel("Densidade de Probabilidade")

plt.title("Curva da Distribuição Normal para UTI\_TOTAL\_EXIST")

plt.legend()

plt.show()

# Média e desvio padrão

mu = np.mean(leitos['UTI\_TOTAL\_SUS'])

sigma = np.std(leitos['UTI\_TOTAL\_SUS'])

# Gerar valores para o eixo x que cubram a maioria dos dados

x = np.linspace(mu - 4\*sigma, mu + 4\*sigma, 1000)

# Calcular a função densidade de probabilidade da distribuição normal

y = (1 / (sigma \* np.sqrt(2 \* np.pi))) \* np.exp(-(x - mu)\*\*2 / (2 \* sigma\*\*2))

# Plotando a curva normal

plt.plot(x, y, label="Curva Normal")

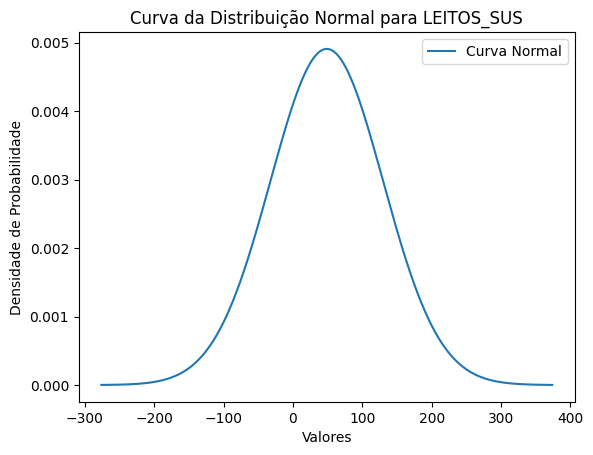
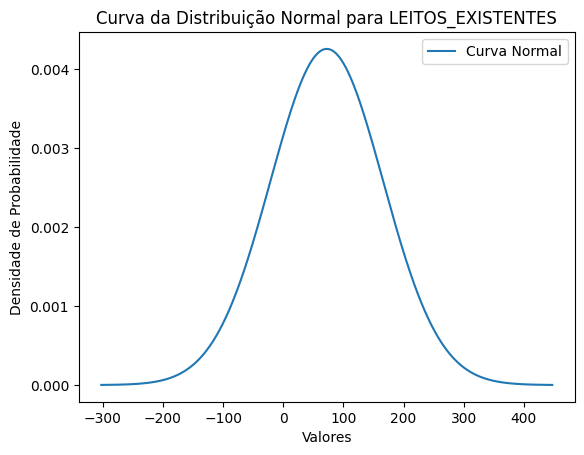
plt.xlabel("Valores")

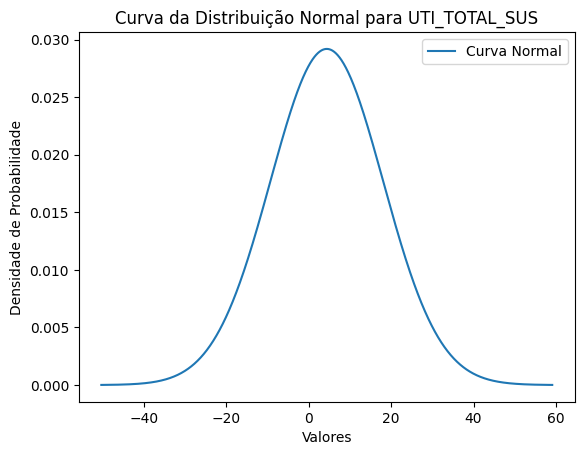
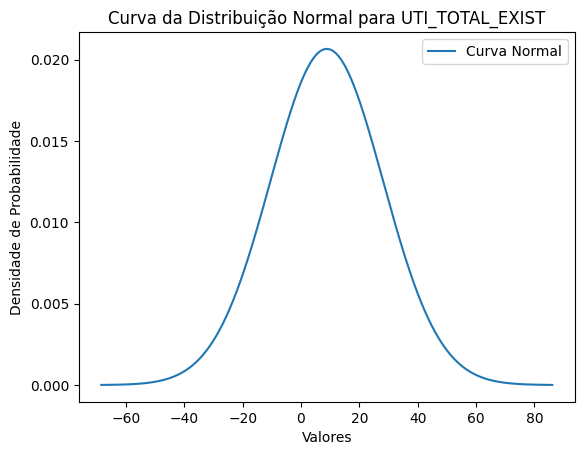
plt.ylabel("Densidade de Probabilidade")

plt.title("Curva da Distribuição Normal para UTI\_TOTAL\_SUS")

plt.legend()

plt.show()

****

****

Os dados foram normalizados, resultando em um novo DataFrame com valores padronizados, onde a média é 0 e o desvio padrão é 1.

**Código da análise:**

#importar a bibliotaca StandardScaler para aplicar a normalização (z=(valor - média)/desvio padão)

#Média = 0 e o Variância = 1

from sklearn.preprocessing import StandardScaler #sklearn.preprocessing fornece várias funções de utilidade comuns e classes de transformadores para alterar vetores de recursos brutos em uma representação mais adequada

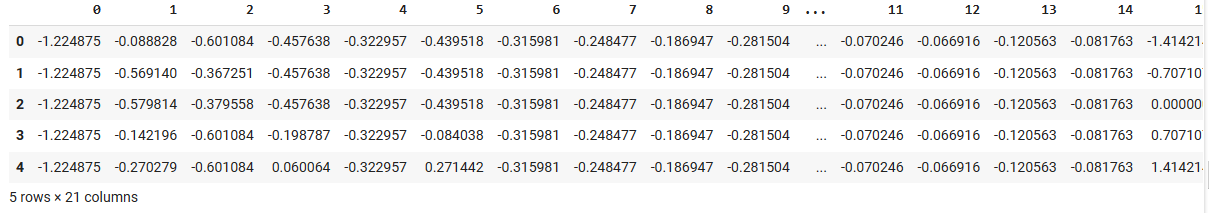
scaler = StandardScaler() #o módulo standard.scaler padroniza os dados

ajuste = scaler.fit(leitos) #determina a média e o desvio padrão / NAO PODE TER COLUNA STRING (TEM CONVERTER A COLUNA)

normal = scaler.transform(leitos) #aplicar a normal padrão z=(valor - média)/desvio padrão

normalpadrao = pd.DataFrame(normal)

normalpadrao.head()



|index|0|1|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16|17|18|19|

|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

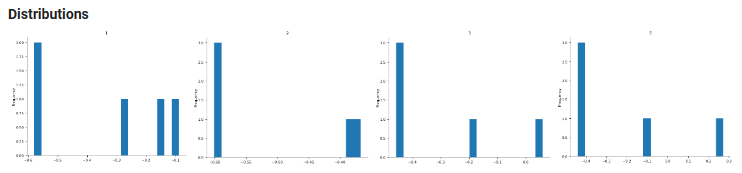
|0|-1\.2248750398813253|-0\.08882803691852455|-0\.6010840661188123|-0\.4576384436960423|-0\.3229573682299073|-0\.43951766409308296|-0\.31598109068805585|-0\.24847709947491162|-0\.1869465299013523|-0\.2815039354905614|-0\.20904680579560914|-0\.07024564237982632|-0\.06691635282619242|-0\.12056297722262184|-0\.0817630651487478|-1\.414213562373095|-1\.6690459207925603|-1\.6996731711975948|0\.4472135954999579|-1\.224744871391589|

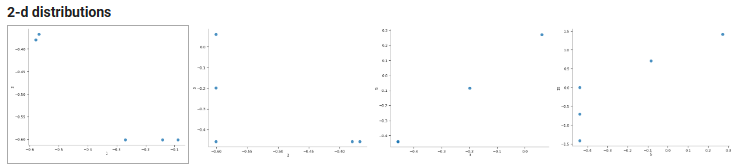
|1|-1\.2248750398813253|-0\.5691400943411691|-0\.36725106534953006|-0\.4576384436960423|-0\.3229573682299073|-0\.43951766409308296|-0\.31598109068805585|-0\.24847709947491162|-0\.1869465299013523|-0\.2815039354905614|-0\.20904680579560914|-0\.07024564237982632|-0\.06691635282619242|-0\.12056297722262184|-0\.0817630651487478|-0\.7071067811865475|-1\.5406577730392865|-1\.6343011261515334|-0\.4472135954999579|0\.0|

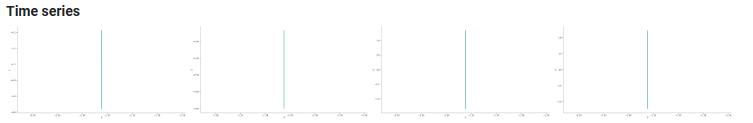
|2|-1\.2248750398813253|-0\.5798136956172278|-0\.37955806539001863|-0\.4576384436960423|-0\.3229573682299073|-0\.43951766409308296|-0\.31598109068805585|-0\.24847709947491162|-0\.1869465299013523|-0\.2815039354905614|-0\.20904680579560914|-0\.07024564237982632|-0\.06691635282619242|-0\.12056297722262184|-0\.0817630651487478|0\.0|-1\.4122696252860127|-1\.5689290811054721|-1\.3416407864998738|1\.224744871391589|

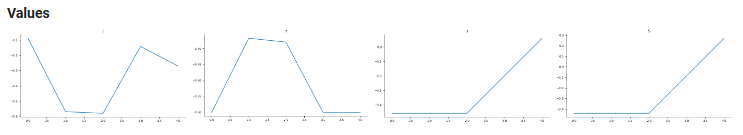
|3|-1\.2248750398813253|-0\.14219604329881838|-0\.6010840661188123|-0\.1987869740682289|-0\.3229573682299073|-0\.0840377814402877|-0\.31598109068805585|-0\.24847709947491162|-0\.1869465299013523|-0\.2815039354905614|-0\.20904680579560914|-0\.07024564237982632|-0\.06691635282619242|-0\.12056297722262184|-0\.0817630651487478|0\.7071067811865475|-1\.2838814775327387|-1\.5035570360594108|1\.3416407864998738|NaN|

|4|-1\.2248750398813253|-0\.2702792586115236|-0\.6010840661188123|0\.060064495559584444|-0\.3229573682299073|0\.27144210121250756|-0\.31598109068805585|-0\.24847709947491162|-0\.1869465299013523|-0\.2815039354905614|-0\.20904680579560914|-0\.07024564237982632|-0\.06691635282619242|-0\.12056297722262184|-0\.0817630651487478|1\.414213562373095|-1\.155493329779465|-1\.4381849910133495|NaN|NaN|









**Código da análise:**

# Cálculo da distribuição de frequências relativas

fp = leitos['UTI\_TOTAL\_EXIST'].value\_counts(sort=False).sort\_index() / leitos['LEITOS\_EXISTENTES'].count()

print("Distribuição de frequências relativas para UTI\_TOTAL\_EXIST:")

print(fp)

# Cálculo da função de repartição (cumulativa)

fp\_cumsum = fp.cumsum()

print("Função de repartição para UTI\_TOTAL\_EXIST:")

print(fp\_cumsum)

# Cálculo da distribuição de frequências relativas

fp = leitos['UTI\_TOTAL\_SUS'].value\_counts(sort=False).sort\_index() / leitos['UTI\_TOTAL\_EXIST'].count()

print("Distribuição de frequências relativas para UTI\_TOTAL\_SUS:")

print(fp)

# Cálculo da função de repartição (cumulativa)

fp\_cumsum = fp.cumsum()

print("Função de repartição para UTI\_TOTAL\_SUS:")

print(fp\_cumsum)

# Cálculo da distribuição de frequências relativas

fp = leitos['LEITOS\_SUS'].value\_counts(sort=False).sort\_index() / leitos['LEITOS\_EXISTENTES'].count()

print("Distribuição de frequências relativas para LEITOS\_SUS:")

print(fp)

# Cálculo da função de repartição (cumulativa)

fp\_cumsum = fp.cumsum()

print("Função de repartição para LEITOS\_SUS:")

print(fp\_cumsum)

Distribuição de frequências relativas para UTI\_TOTAL\_EXIST:

UTI\_TOTAL\_EXIST

0 0.660353

1 0.003070

2 0.002645

3 0.001606

4 0.004015

...

235 0.000142

243 0.000142

245 0.000047

297 0.000094

380 0.000142

Name: count, Length: 134, dtype: float64

Função de repartição para UTI\_TOTAL\_EXIST:

UTI\_TOTAL\_EXIST

0 0.660353

1 0.663423

2 0.666068

3 0.667674

4 0.671689

...

235 0.999575

243 0.999717

245 0.999764

297 0.999858

380 1.000000

Name: count, Length: 134, dtype: float64

Distribuição de frequências relativas para UTI\_TOTAL\_SUS:

UTI\_TOTAL\_SUS

0 0.805592

1 0.000850

2 0.002551

3 0.001559

4 0.003826

...

180 0.000142

194 0.000142

225 0.000142

245 0.000142

365 0.000142

Name: count, Length: 102, dtype: float64

Função de repartição para UTI\_TOTAL\_SUS:

UTI\_TOTAL\_SUS

0 0.805592

1 0.806442

2 0.808993

3 0.810552

4 0.814377

...

180 0.999433

194 0.999575

225 0.999717

245 0.999858

365 1.000000

Name: count, Length: 102, dtype: float64

Distribuição de frequências relativas para LEITOS\_SUS:

LEITOS\_SUS

0 0.239278

1 0.014359

2 0.008974

3 0.006896

4 0.007368

...

988 0.000142

1046 0.000142

1078 0.000142

1153 0.000142

1476 0.000142

Name: count, Length: 404, dtype: float64

Função de repartição para LEITOS\_SUS:

LEITOS\_SUS

0 0.239278

1 0.253637

2 0.262611

3 0.269507

4 0.276875

...

988 0.999433

1046 0.999575

1078 0.999717

1153 0.999858

1476 1.000000

Name: count, Length: 404, dtype: float64

A normalização dos dados é uma etapa crucial para muitas técnicas de análise de dados, especialmente aquelas que assumem que os dados seguem uma distribuição normal. A transformação z-score facilita a comparação entre diferentes variáveis e é essencial para algoritmos de machine learning que são sensíveis à escala dos dados.

Os dados de ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’, ‘UTI\_TOTAL\_SUS’ foram normalizados e analisados para verificar sua distribuição. A curva normal ajustada sugere que os dados podem seguir uma distribuição normal, enquanto a normalização z-score prepara os dados para análises subsequentes.

A distribuição de frequências relativas e a função de repartição foram calculadas para ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’, ‘UTI\_TOTAL\_SUS’, fornecendo insights adicionais sobre a distribuição dos dados dentro do dataset.

# 

# **6. Distribuição Amostral**

Aqui detalha a criação de distribuições amostrais para as variáveis ‘LEITOS\_EXISTENTES’, ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’ e ‘UTI\_TOTAL\_SUS’ do dataset ‘Leitos2024’. O objetivo é observar como a distribuição amostral da média se comporta com o aumento do tamanho da amostra.

**Metodologia:**

A análise foi realizada utilizando a biblioteca Numpy da linguagem de programação Python, com foco na geração de médias amostrais para diferentes tamanhos de amostra. Utilizamos a função np.random.choice para selecionar amostras aleatórias e a função np.mean para calcular a média dessas amostras. O processo foi repetido 1000 vezes para cada tamanho de amostra escolhido.

**Resultados:**

* **Distribuições Amostrais:** Foram geradas distribuições amostrais para os tamanhos de amostra de 1, 2, 10, 50 e 100. Cada distribuição foi visualizada em um histograma, permitindo a observação da forma da distribuição e sua aproximação à normalidade conforme o tamanho da amostra aumentava.

**Código do gráfico:**

#importar as libs para trabalhar com números aleatórios

from numpy.random import seed #inicializar o gerador de números aleatórios

from numpy.random import randint #retorna um número inteiro aleatório

from numpy import mean

# Inicializar o gerador de números aleatórios para reprodutibilidade

np.random.seed(1)

# Função para calcular as médias amostrais

def calcular\_medias\_amostrais(data, tamanho\_amostra, repeticoes):

medias\_amostrais = [np.mean(np.random.choice(data, tamanho\_amostra)) for \_ in range(repeticoes)]

return medias\_amostrais

# Variáveis do dataset

variaveis = ['LEITOS\_EXISTENTES', 'LEITOS\_SUS', 'UTI\_TOTAL\_EXIST', 'UTI\_TOTAL\_SUS']

# Loop para cada variável

for variavel in variaveis:

dados = leitos[variavel].values

for tamanho\_amostra in [1, 2, 10, 50, 100]:

medias\_amostrais = calcular\_medias\_amostrais(dados, tamanho\_amostra, 1000)

plt.hist(medias\_amostrais, bins=30, edgecolor='black')

plt.title(f'Distribuição das Médias Amostrais de {variavel} (n={tamanho\_amostra})')

plt.xlabel('Média Amostral')

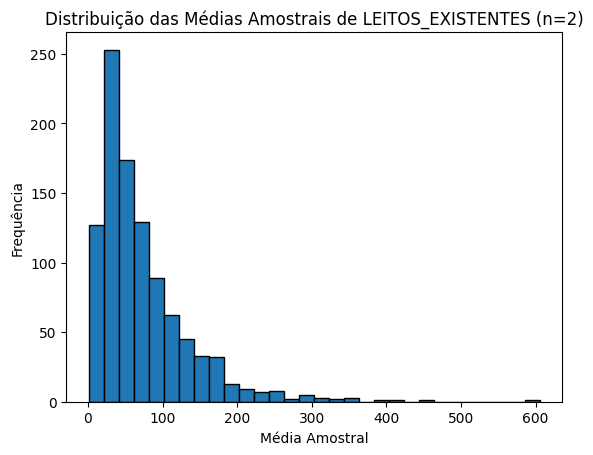
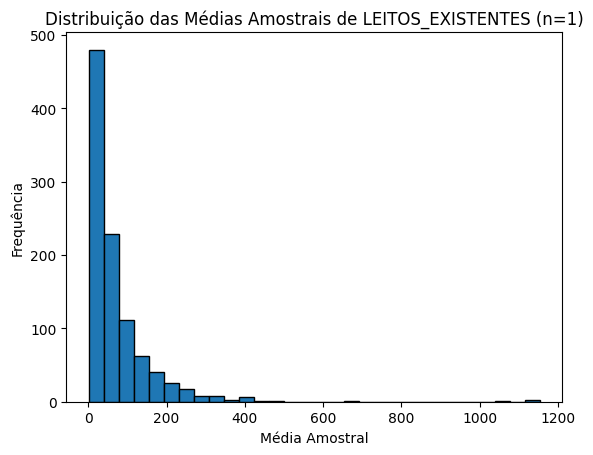
plt.ylabel('Frequência')

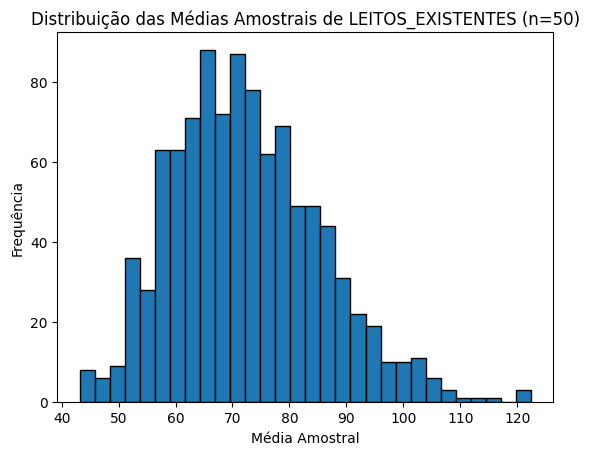
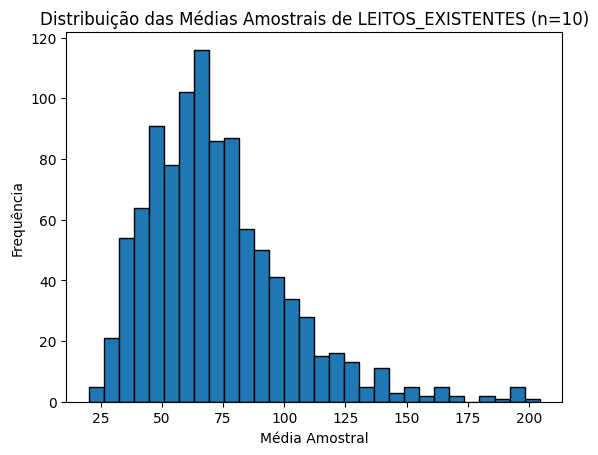
plt.show()

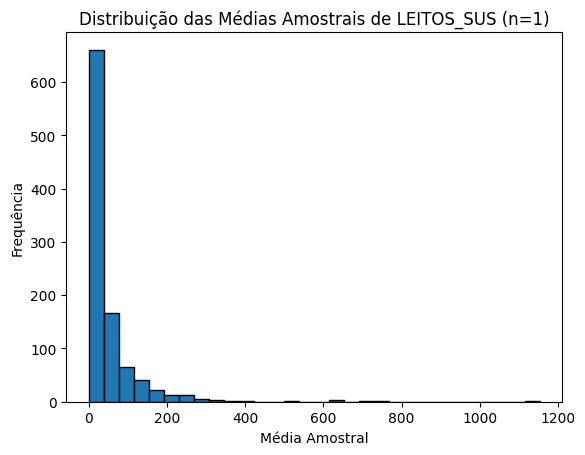
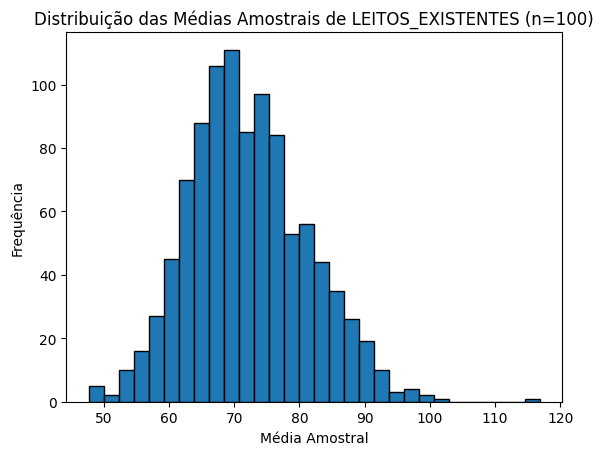
Os resultados mostraram que, para tamanhos de amostra pequenos, a distribuição das médias amostrais tende a seguir a distribuição dos dados originais. À medida que o tamanho da amostra aumenta, a distribuição das médias amostrais se aproxima de uma distribuição normal, em conformidade com o Teorema do Limite Central.

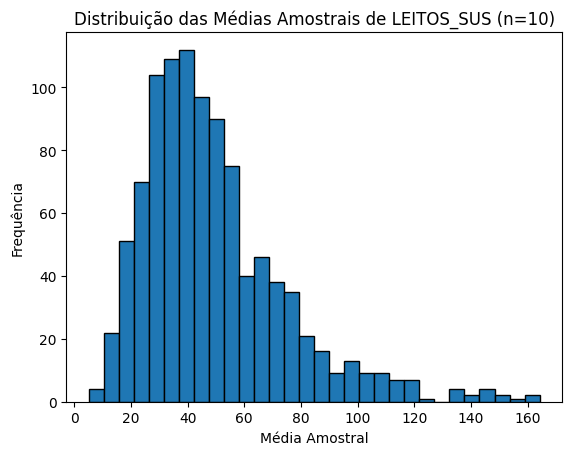
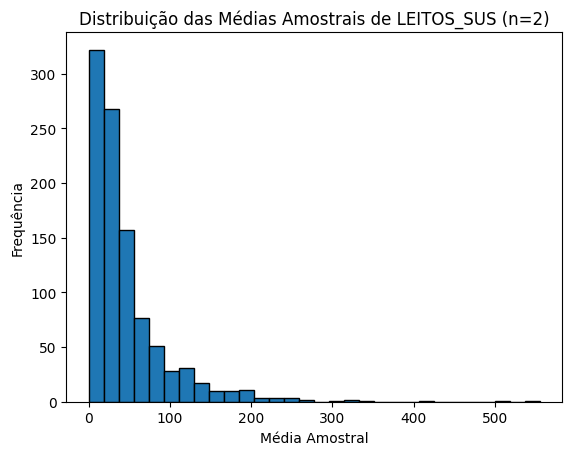
A análise das distribuições amostrais reforça a importância do tamanho da amostra na estimação de parâmetros populacionais. Os histogramas indicam que, com o aumento do tamanho da amostra, a média amostral se torna um estimador mais confiável da média populacional.

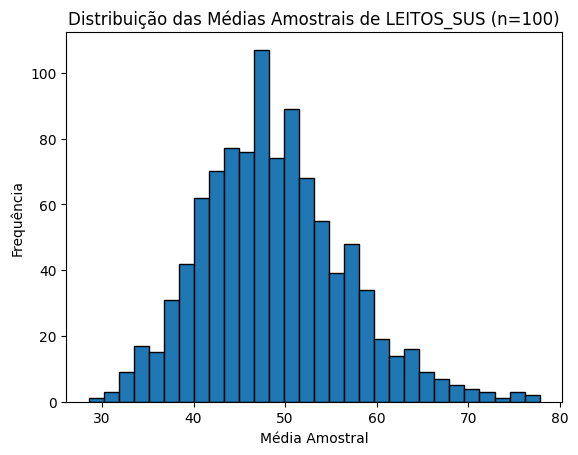
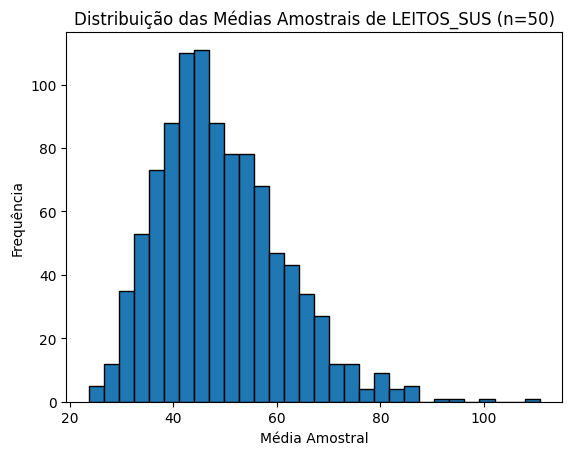
A análise foi replicada para outras variáveis relevantes do dataset, demonstrando a aplicabilidade do método em diferentes contextos dentro dos dados de leitos e UTIs.

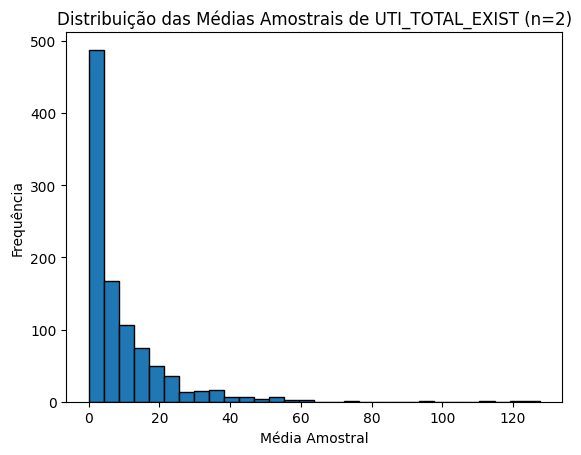
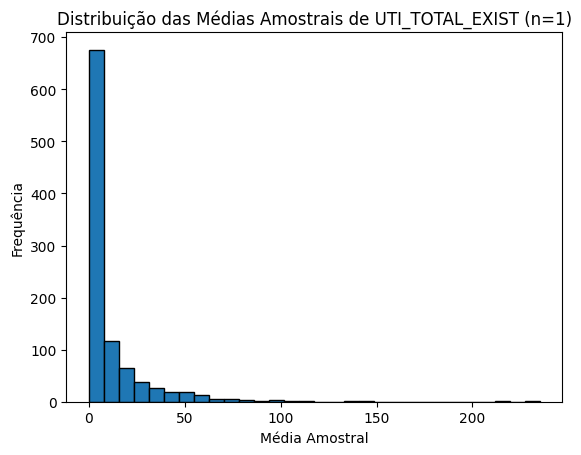


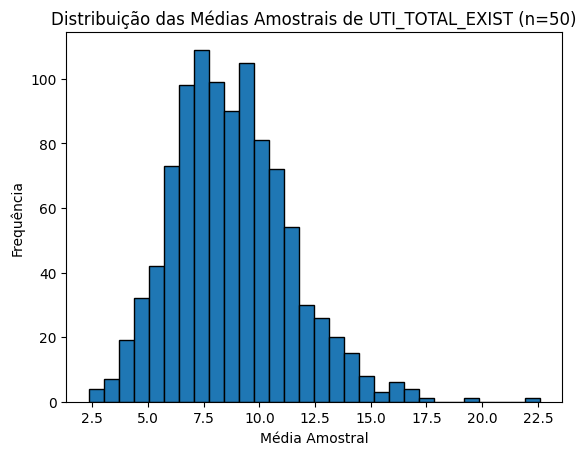
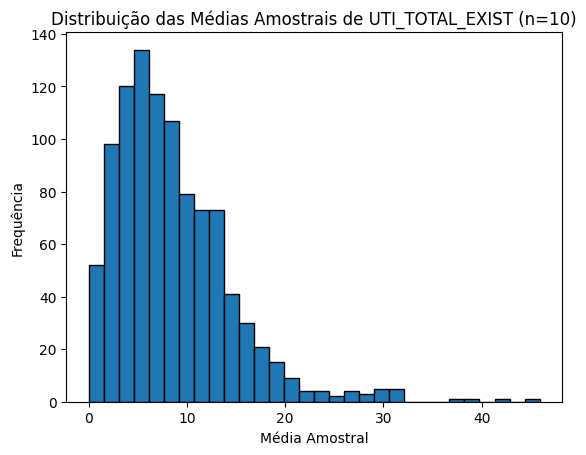


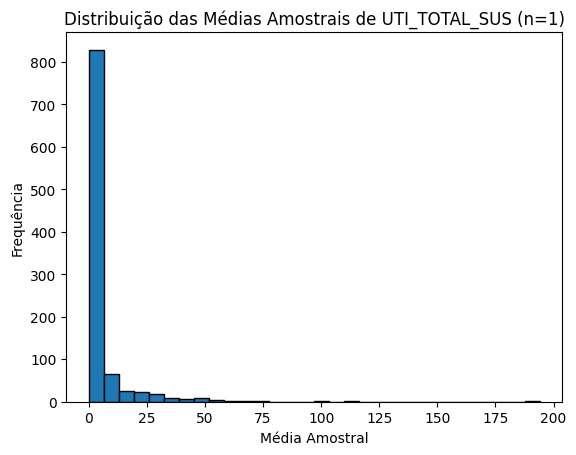
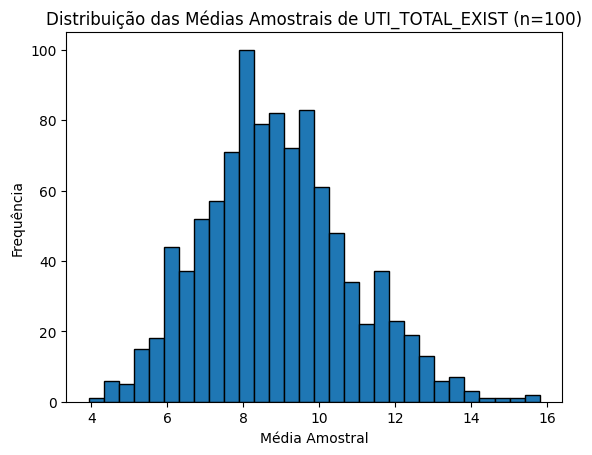


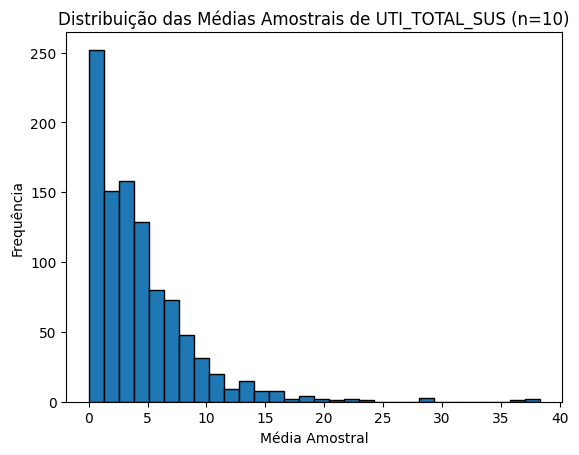
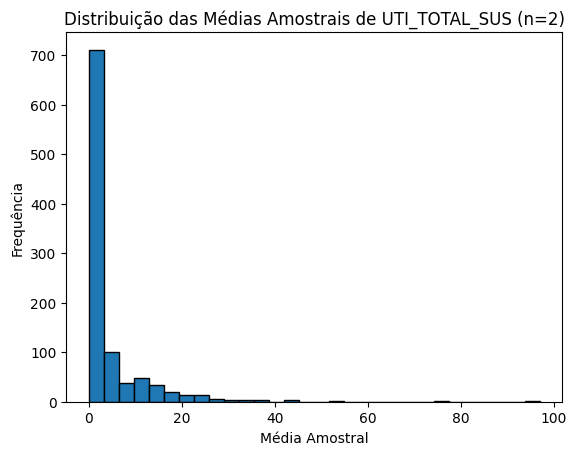


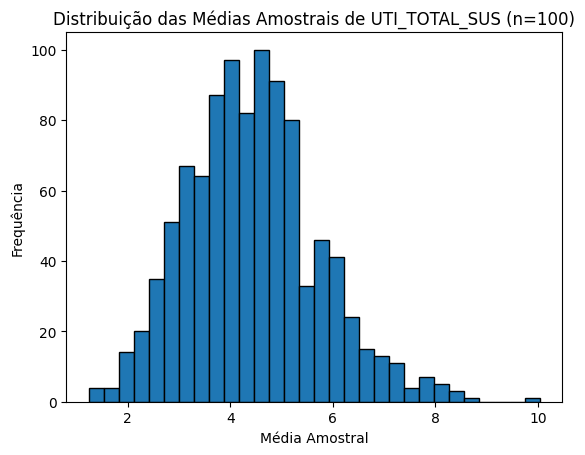
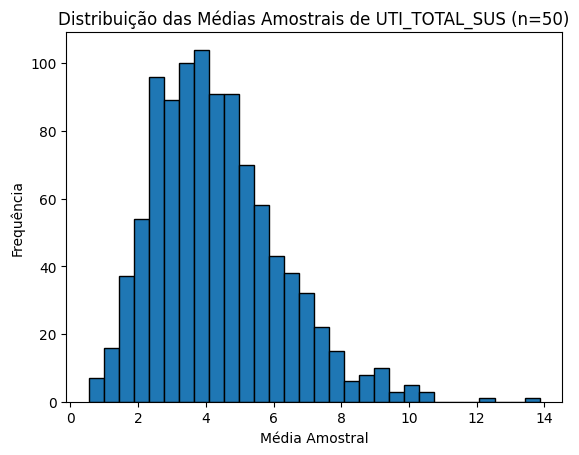










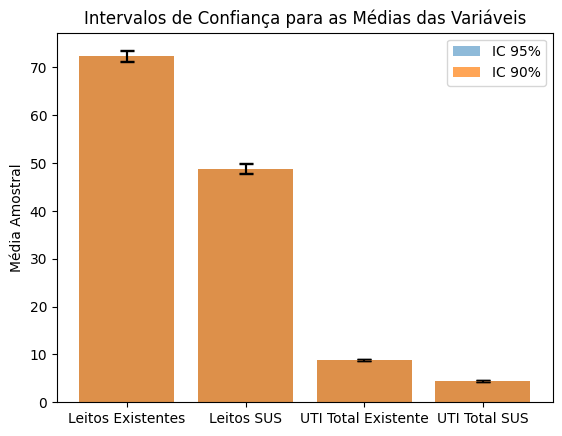


# **7. Intervalo de Confiança**

A análise dos intervalos de confiança (IC) forneceu uma estimativa de onde a verdadeira média populacional pode estar com base numa amostra. Os IC’s foram calculados para as variáveis ‘LEITOS\_EXISTENTES’, ‘LEITOS\_SUS’, ‘UTI\_TOTAL\_EXIST’ e ‘UTI\_TOTAL\_SUS’ do dataset ‘Leitos2024’, com níveis de confiança de 90% e 95%.

**Resultados:**

* **LEITOS\_EXISTENTES:**
  + IC de 90%: (71.26, 73.38)
  + IC de 95%: (71.06, 73.58) Estes intervalos sugerem que, com um nível de confiança de 90% e 95%, a média verdadeira de leitos existentes está estimada dentro desses intervalos.
* **LEITOS\_SUS:**
  + IC de 90%: (47.92, 49.76)
  + IC de 95%: (47.75, 49.94) Similarmente, para leitos SUS, os intervalos indicam onde a média verdadeira está provavelmente localizada, com os respectivos níveis de confiança.
* **UTI\_TOTAL\_EXIST:**
  + IC de 90%: (8.62, 9.06)
  + IC de 95%: (8.58, 9.10) Para o total de UTIs existentes, os intervalos refletem a estimativa da média com uma margem de erro menor para o IC de 95%.
* **UTI\_TOTAL\_SUS:**
  + IC de 90%: (4.26, 4.57)
  + IC de 95%: (4.23, 4.60) Para o total de UTIs SUS, os intervalos são consistentes com os outros ICs, mostrando onde a média populacional é estimada para estar.

****

**Intervalo de Confiança de 90% para LEITOS\_EXISTENTES:**

**(71.26305063779212, 73.38138540981792)**

**Intervalo de Confiança de 95% para LEITOS\_EXISTENTES:**

**(71.06012531034521, 73.58431073726483)**

**Intervalo de Confiança de 90% para LEITOS\_SUS:**

**(47.92223399818023, 49.75942101787871)**

**Intervalo de Confiança de 95% para LEITOS\_SUS:**

**(47.746241146400244, 49.935413869658696)**

**Intervalo de Confiança de 90% para UTI\_TOTAL\_EXIST:**

**(8.621417409275104, 9.058159390271467)**

**Intervalo de Confiança de 95% para UTI\_TOTAL\_EXIST:**

**(8.579579825020778, 9.099996974525792)**

**Intervalo de Confiança de 90% para UTI\_TOTAL\_SUS:**

**(4.259296029210327, 4.5683064646494875)**

**Intervalo de Confiança de 95% para UTI\_TOTAL\_SUS:**

**(4.229694455075736, 4.5979080387840785)**

Os intervalos de confiança mais amplos para um nível de confiança de 95% refletem maior certeza de que a média populacional verdadeira está dentro desses limites. A diferença entre os ICs de 90% e 95% é relativamente pequena, indicando que a média amostral é uma boa estimativa da média populacional e que os dados são relativamente precisos.

Os intervalos de confiança calculados fornecem uma visão valiosa sobre a variabilidade e a precisão das estimativas das médias populacionais. Eles são essenciais para a tomada de decisões informadas e planejamento no contexto do sistema de saúde.

**Código da Análise:**

import numpy as np

import scipy.stats as stats

# Função para calcular o intervalo de confiança

def calcular\_intervalo\_confianca(data, confianca):

media\_amostral = np.mean(data)

desvio\_padrao\_amostral = np.std(data, ddof=1)

tamanho\_amostra = len(data)

t\_critico = stats.t.ppf((1 + confianca) / 2, df=tamanho\_amostra-1)

margem\_erro = t\_critico \* (desvio\_padrao\_amostral / np.sqrt(tamanho\_amostra))

return (media\_amostral - margem\_erro, media\_amostral + margem\_erro)

# Variáveis do dataset

variaveis = ['LEITOS\_EXISTENTES', 'LEITOS\_SUS', 'UTI\_TOTAL\_EXIST', 'UTI\_TOTAL\_SUS']

# Calculando o intervalo de confiança para cada variável

for variavel in variaveis:

dados = leitos[variavel].dropna().values # Removendo valores nulos

IC\_90 = calcular\_intervalo\_confianca(dados, 0.90)

IC\_95 = calcular\_intervalo\_confianca(dados, 0.95)

print(f"Intervalo de Confiança de 90% para {variavel}: {IC\_90}")

print(f"Intervalo de Confiança de 95% para {variavel}: {IC\_95}")

**Código do gráfico:**

import numpy as np

import matplotlib.pyplot as plt

import scipy.stats as stats

# Função para calcular o intervalo de confiança e a margem de erro

def calcular\_intervalo\_confianca(data, confianca):

media\_amostral = np.mean(data)

desvio\_padrao\_amostral = np.std(data, ddof=1)

tamanho\_amostra = len(data)

t\_critico = stats.t.ppf((1 + confianca) / 2, df=tamanho\_amostra-1)

margem\_erro = t\_critico \* (desvio\_padrao\_amostral / np.sqrt(tamanho\_amostra))

return media\_amostral, margem\_erro

# Variáveis do dataset e seus respectivos nomes para o gráfico

variaveis = ['LEITOS\_EXISTENTES', 'LEITOS\_SUS', 'UTI\_TOTAL\_EXIST', 'UTI\_TOTAL\_SUS']

nomes\_variaveis = ['Leitos Existentes', 'Leitos SUS', 'UTI Total Existente', 'UTI Total SUS']

# Listas para armazenar os resultados

medias = []

margens\_erro\_90 = []

margens\_erro\_95 = []

# Calculando o intervalo de confiança e a margem de erro para cada variável

for variavel in variaveis:

dados = leitos[variavel].dropna().values # Removendo valores nulos

media\_amostral, margem\_erro\_90 = calcular\_intervalo\_confianca(dados, 0.90)

\_, margem\_erro\_95 = calcular\_intervalo\_confianca(dados, 0.95)

medias.append(media\_amostral)

margens\_erro\_90.append(margem\_erro\_90)

margens\_erro\_95.append(margem\_erro\_95)

# Criando o gráfico de barras de erro

posicoes = np.arange(len(nomes\_variaveis))

plt.bar(posicoes, medias, yerr=margens\_erro\_95, capsize=5, alpha=0.5, label='IC 95%')

plt.bar(posicoes, medias, yerr=margens\_erro\_90, capsize=5, alpha=0.7, label='IC 90%')

plt.xticks(posicoes, nomes\_variaveis)

plt.ylabel('Média Amostral')

plt.title('Intervalos de Confiança para as Médias das Variáveis')

plt.legend()

plt.show()

# 

# **8. Conclusão**

A análise compreensiva do dataset Leitos2024, abrangendo desde a estatística descritiva até a inferência estatística, proporcionou **insights valiosos** sobre a distribuição e gestão de leitos e UTIs no Brasil. A metodologia aplicada revelou não apenas a situação atual dos recursos hospitalares, mas também forneceu ferramentas para previsões e planejamento estratégico.

**Síntese dos Métodos e Resultados:**

* **Estatística Descritiva:** Foi essencial para entender a distribuição geral dos dados, identificar outliers e fornecer um resumo quantitativo das variáveis analisadas.
* **Normal Padrão:** Confirmou a aplicabilidade da distribuição normal, permitindo comparações e análises mais precisas entre as variáveis.
* **Distribuições Amostrais:** Reforçou o Teorema do Limite Central, indicando que as médias amostrais tendem a uma distribuição normal à medida que o tamanho da amostra aumenta.
* **Intervalo de Confiança:** Ofereceu uma estimativa da precisão e variabilidade das médias populacionais, essenciais para inferências estatísticas.

**Visualização de Dados:**

As visualizações, incluindo box plots, histogramas, scatter plots e mapas de calor, foram cruciais para explorar visualmente as relações entre variáveis e entender a estrutura dos dados. Os mapas de calor de correlação destacaram as inter-relações entre diferentes tipos de leitos e UTIs, servindo como base para decisões informadas e alocação eficiente de recursos.

**Implicações Práticas e Recomendações:**

A análise ofereceu uma visão valiosa para os gestores de saúde, permitindo uma melhor compreensão da capacidade e demanda de leitos e UTIs. Recomenda-se a realização de análises adicionais para explorar outras variáveis do dataset e aplicar modelos preditivos, visando mais previsões para a gestão de leitos e UTIs. Além disso, a análise de tendências ao longo do tempo e comparações entre diferentes regiões ou instituições podem revelar padrões importantes e oportunidades de melhoria no sistema de saúde.

**Insights:**

Com base nas análises realizadas, pudemos identificar que a **região Nordeste**, apesar de ter uma quantidade significativa de leitos e UTIs existentes e destinados ao SUS, apresenta uma **média relativamente baixa de leitos por habitante**. Essa média é inferior à das outras regiões, como Centro-Oeste e Norte.

A distribuição desigual de leitos e UTIs impactam no acesso aos serviços de saúde, especialmente em situações de alta demanda, como epidemias ou emergências. Por isso, é muito importante que as autoridades de saúde avaliem esses dados e busquem estratégias para melhorar a distribuição e garantir um atendimento adequado à população.

Para melhorar a situação, é importante que as autoridades de saúde avaliem a alocação de leitos e busquem estratégias para otimizar a distribuição, garantindo um atendimento adequado a toda a população. Além disso, investimentos contínuos na expansão e manutenção desses recursos são essenciais para enfrentar os desafios de saúde pública na região Nordeste.

**Considerações Finais:**

O relatório técnico demonstra a importância de uma análise estatística rigorosa e visualizações de dados eficazes no contexto da saúde. As técnicas utilizadas proporcionaram uma compreensão mais profunda dos dados, destacando a relevância da estatística descritiva, normalização, distribuições amostrais e intervalos de confiança na análise de dados de saúde.

É de suma importância monitorar esses dados para garantir o acesso adequado aos serviços de saúde durante a pandemia e além.

Essas análises refletem os totais de leitos e UTI, com base na população estimada em cada região. E esses dados são essenciais para compreender a capacidade de atendimento hospitalar na região