Relatório 5- Estatística p/ Aprendizado de Máquina (I)

Felipe Fonseca

Descrição da atividade

Nesse card foi trabalhado a seção de estatísticas, probabilidade e prática em python, do curso de Machine Learning e Data Science.

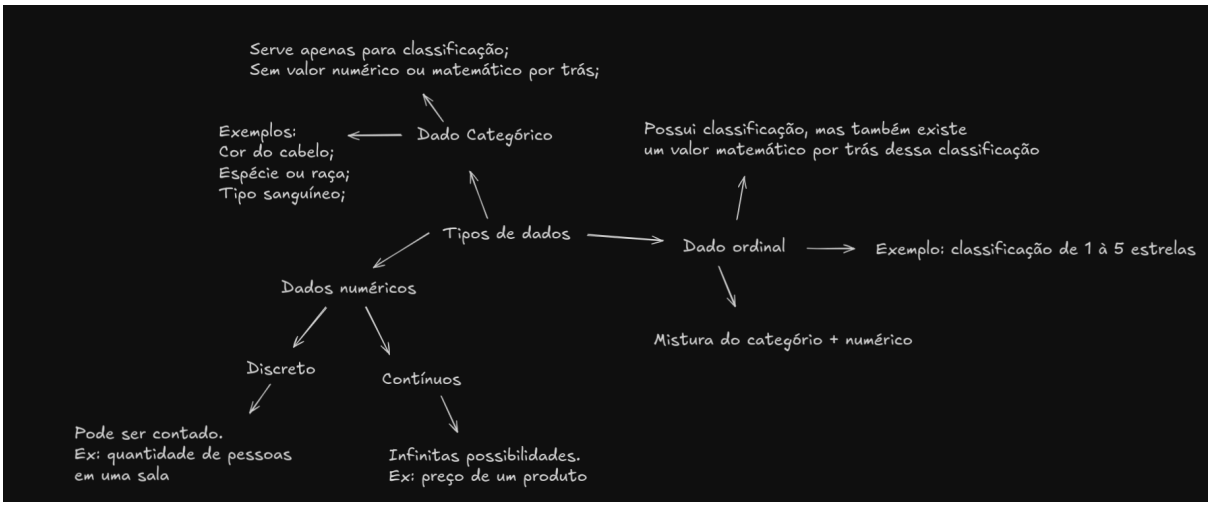
Tipos de dados.

No primeiro vídeo foi ensinado sobre alguns tipos de data que iremos trabalhar, que no caso foram Numerical Data (Dados numéricos), Categorical Data(Dados categóricos) e Ordinal Data(Dados ordinais).

Dados numéricos é aquele que representa um tamanho quantitativo, que pode ser contado, pode ser um número de pessoas, altura de uma pessoa, quantidade de páginas em um livro, preço de um produto etc. Dentro da categoria de dados numéricos tem duas subcategorias: dados discretos e dados contínuos. Dados discretos são os dados numéricos que só podem ser representados por números inteiros, por exemplo quantidade de pessoas, não se pode ter 3,5 pessoas por exemplo. Enquanto os dados contínuos é aquele que contém infinitas possibilidades de valores, por exemplo o preço de um produto.

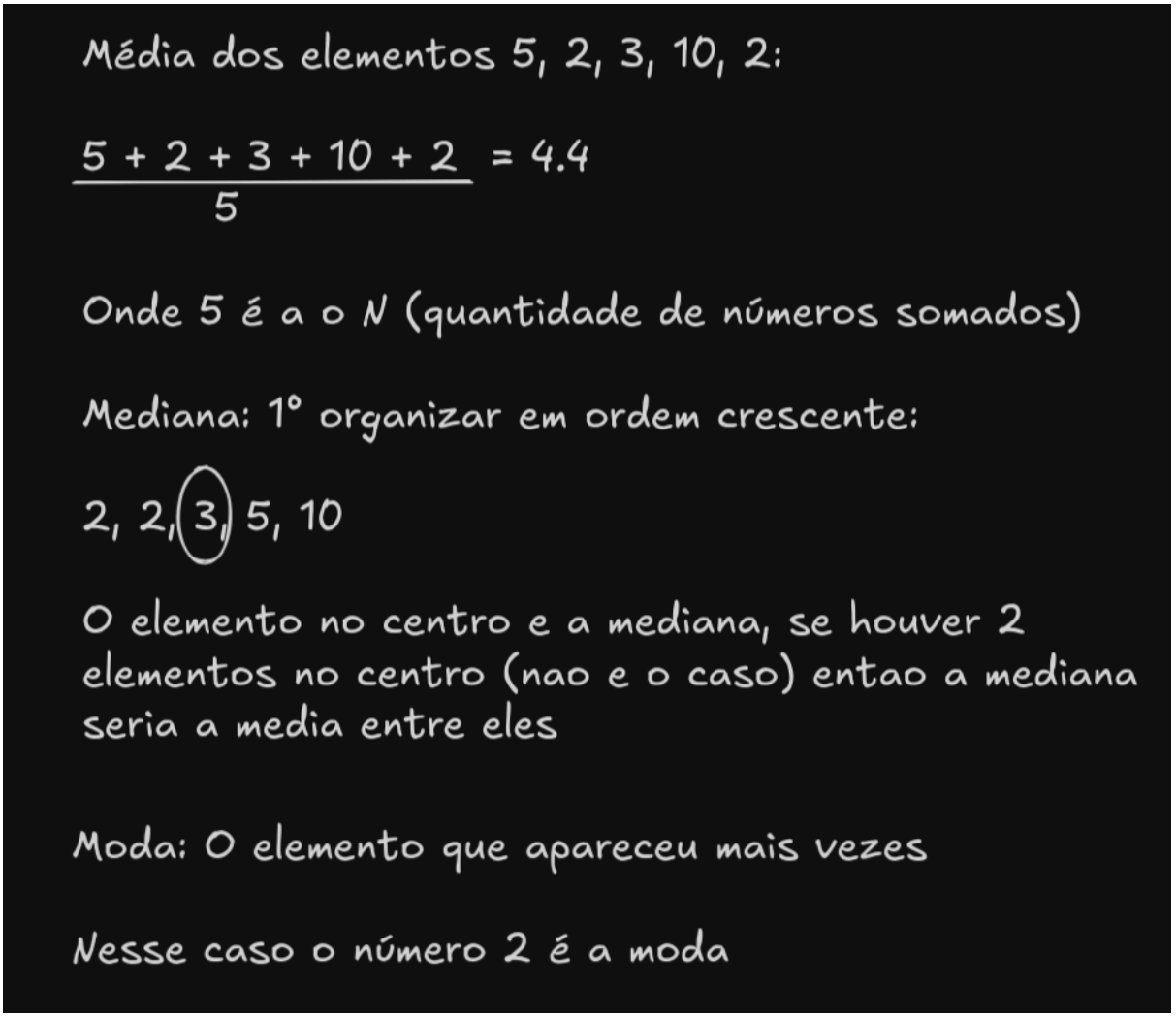
Dados categóricos é aquele que é apenas classificado, por exemplo raça, é apenas uma categoria, não existe um valor que se baseia em cada raça ou algo do tipo, serve apenas para classificação mesmo.

Dados ordinais são aqueles que misturam os dados categóricos e os dados numéricos. O mais comum exemplo seria uma avaliação de 1 à 5 estrelas, onde temos as 5 categorias porém existem valor matemático para eles, é possível dizer que uma categoria é melhor que a outra.



Média, moda e mediana.

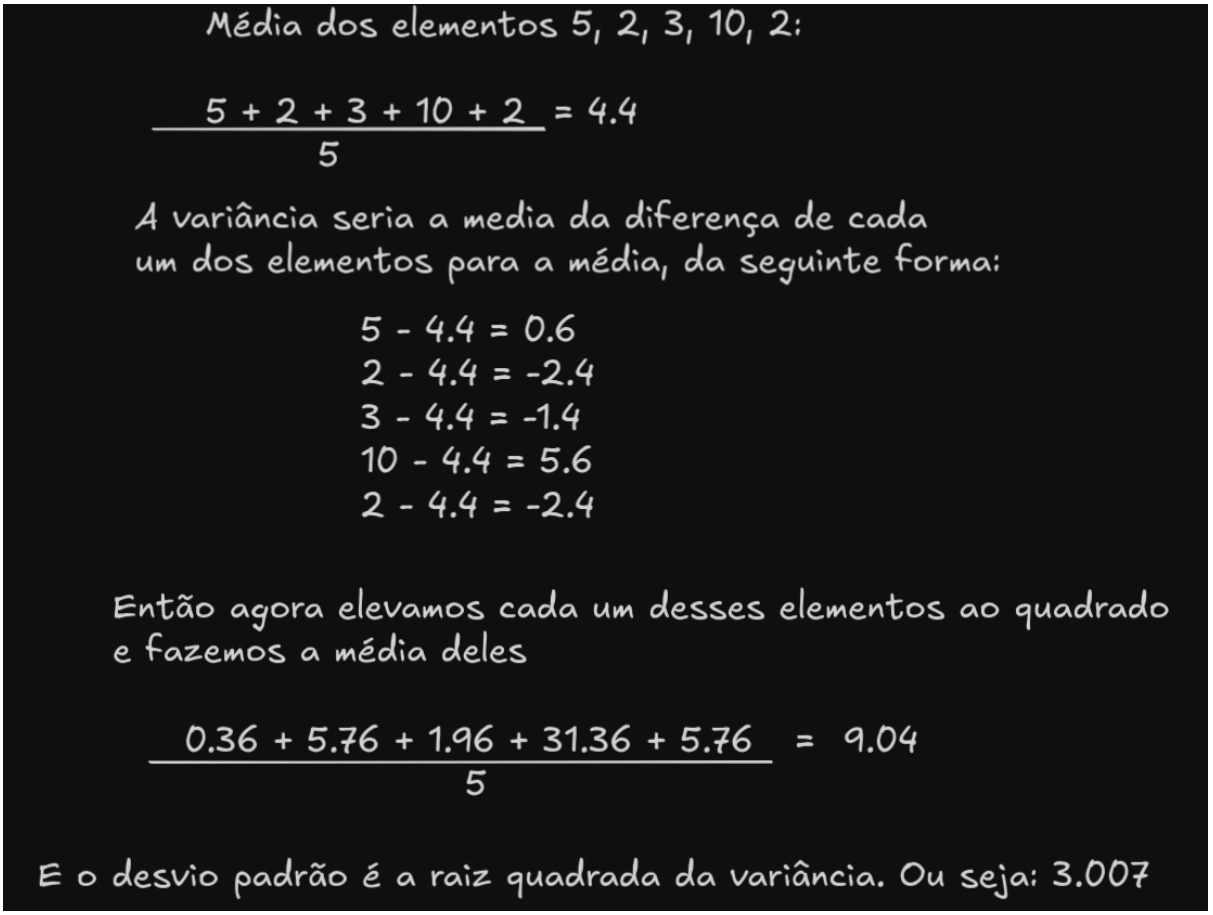
No segundo vídeo o tema era Média, Moda e Mediana. Esse vídeo realmente não serviu de muita coisa pois é um conteúdo ensinado na escola mas em resumo: Média é a soma de todos os valores e somado pela quantidade de valores que foram somados, mediana é quando você coloca todos os valores em ordem crescente e pega o que está exatamente no meio (se a quantidade de valores for par, então a média entre os dois números que estão no meio), e moda é o valor que mais aparece entre a lista dos valores.



Depois tivemos uma atividade mostrando na prática o uso. Através da biblioteca numpy podemos fazer a média e a mediana através dos comandos mean() e do comando median(), e através da biblioteca scipy podemos calcular a média com o comando mode().

Variância.

No próximo vídeo estudamos sobre variância e desvio padrão. Variância é basicamente a média da diferença dos valores para a média dos valores. Enquanto o desvio padrão é apenas a raíz quadrada da variância. Um ponto importante é que quando estamos calculando a variância, para conseguir o resultado precisamos elevar as diferenças ao quadrado para depois dividir pela quantidade, porém ao fazer isso, a medida em que trabalhamos é alterada, por exemplo, se estamos calculando a variância de cm, o resultado da conta já vai ser na medida de cm^2, por isso o desvio padrão é importante, pois ele volta para a medida anterior (nesse caso cm).



Na prática é bem fácil de usar, utilizando a biblioteca numpy pode-se calcular o desvio padrão com std() e a variância com var().

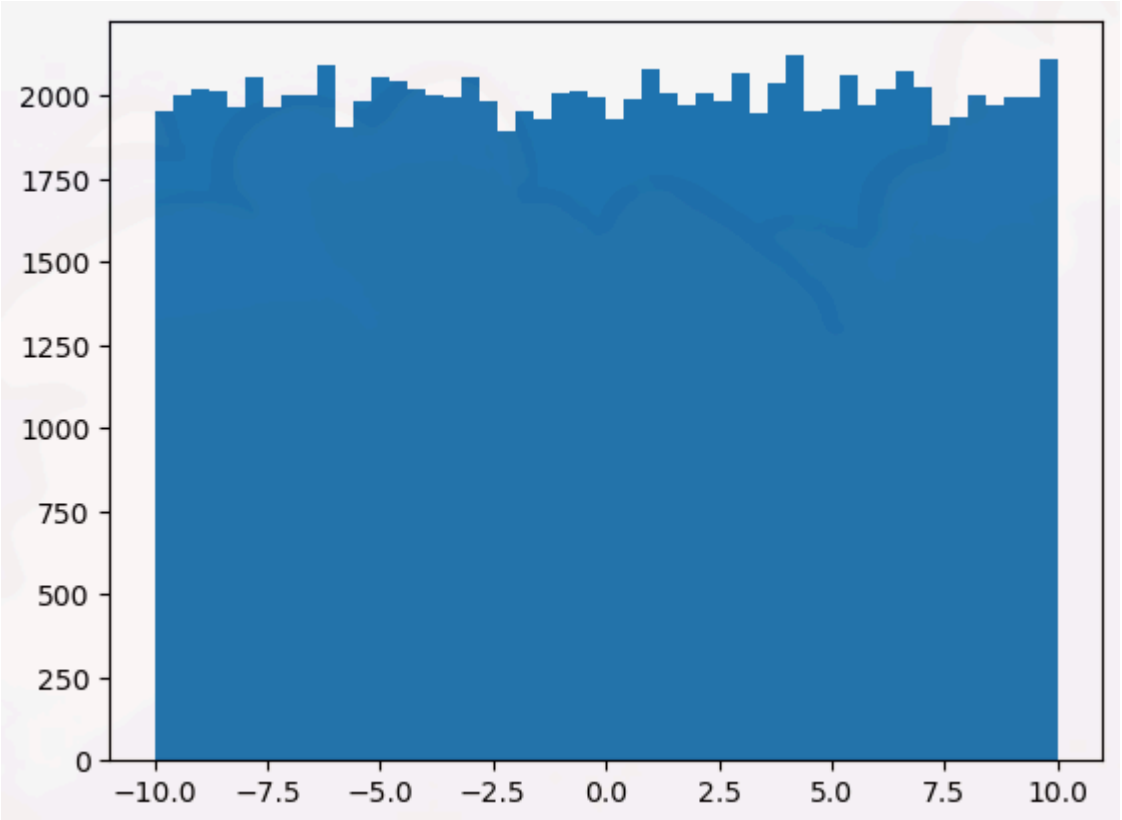
Função de densidade de probabilidade

Função de densidade de probabilidade é uma função que te dá a probabilidade de um dado estar em um espaço dado um determinado valor. É utilizada em valores contínuos como altura, tempo etc.

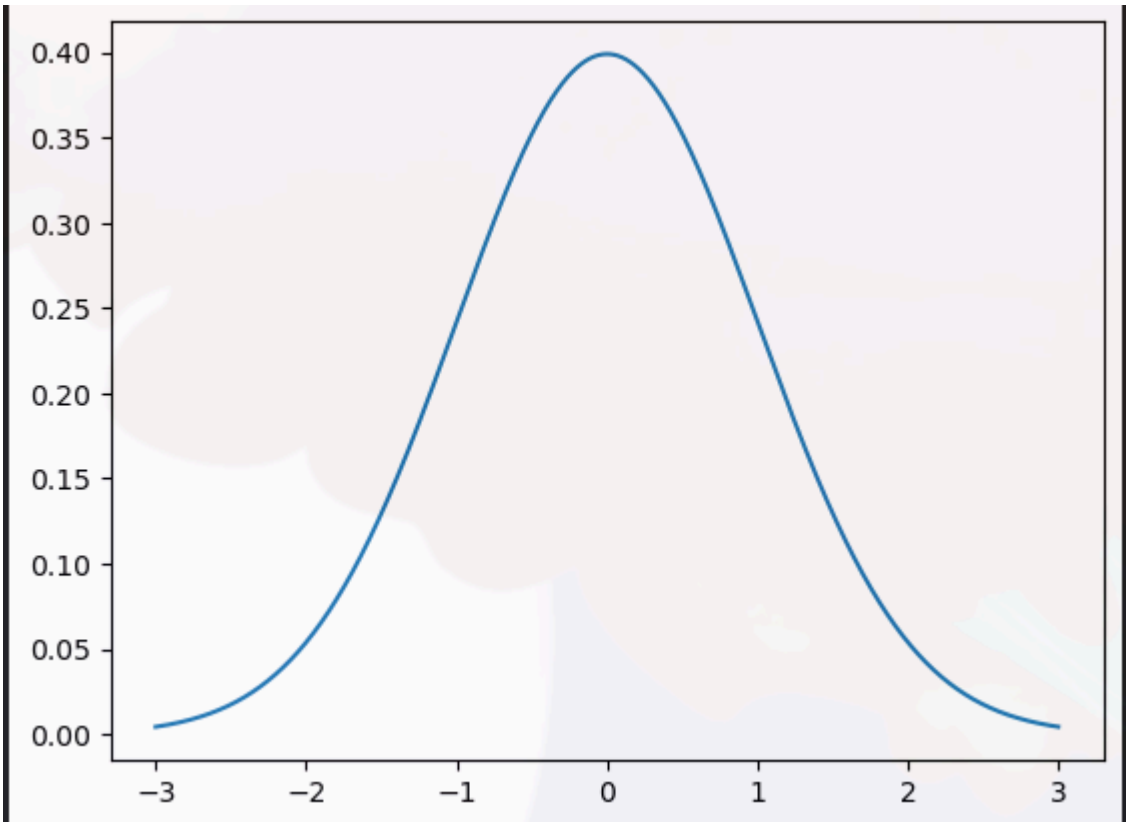
Função de massa de probabilidade é usada quando os valores são contáveis, como lançar um dado, número de pessoas em um lugar etc. Essa função da a probabilidade exata de uma variável.

Exemplo de distribuições de dados.

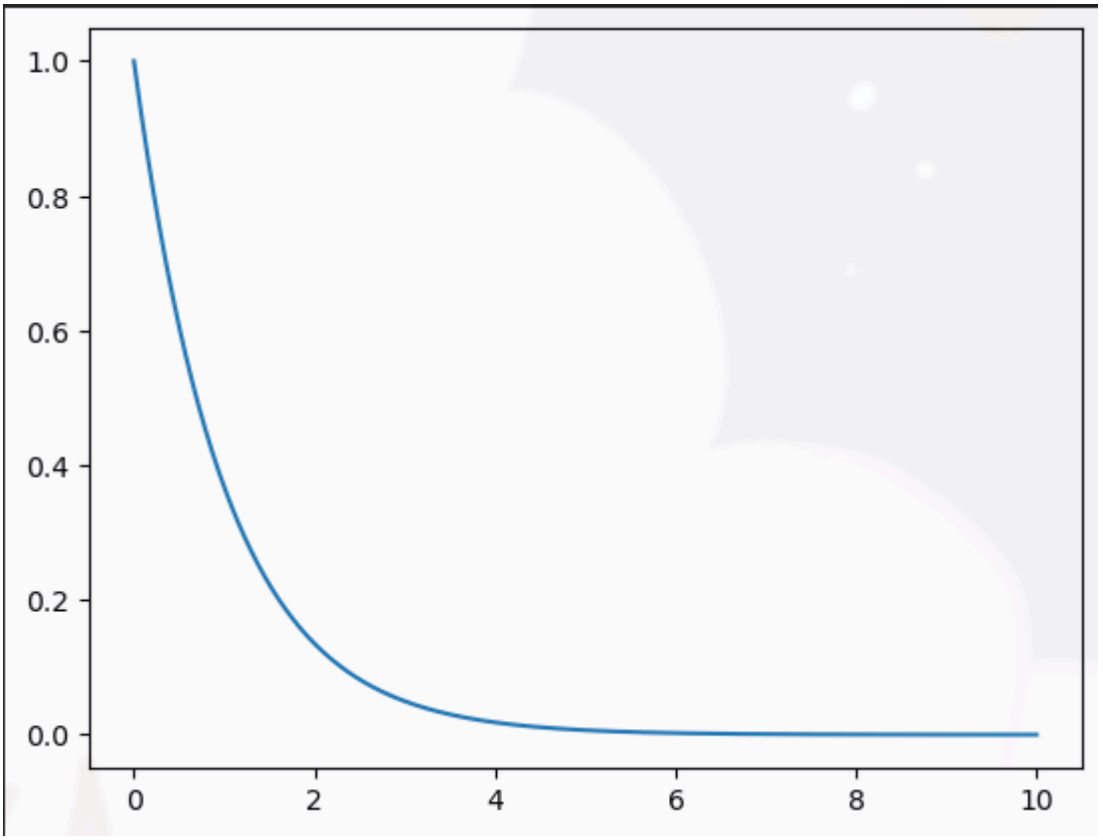
O primeiro exemplo foi uma distribuição uniforme, ou seja, todos os dados têm a mesma chance de aparecerem.



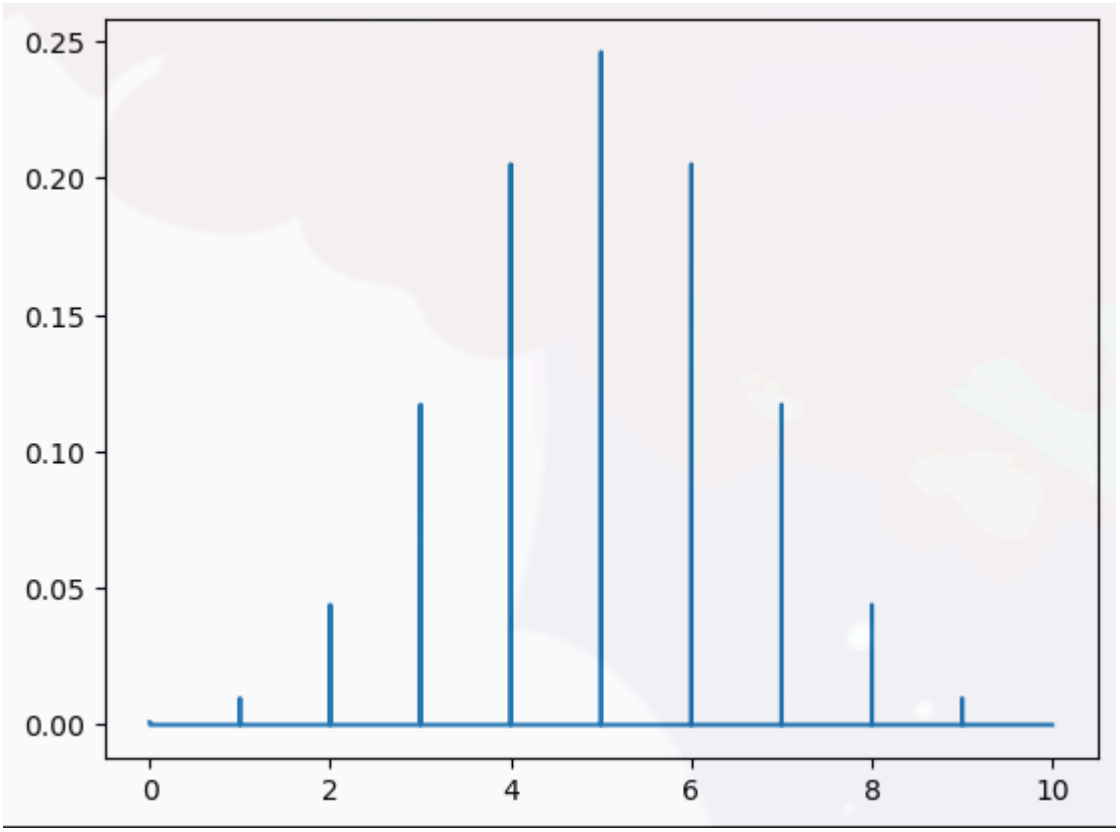
Distribuição normal é quando a maior chance de aparição pertence a média, e quanto mais distante da média menor a chance de aparecer



A distribuição exponencial se comporta de forma que a maior chance é no início, e então temos uma queda exponencial dos valores, quanto mais distante menor a chance.



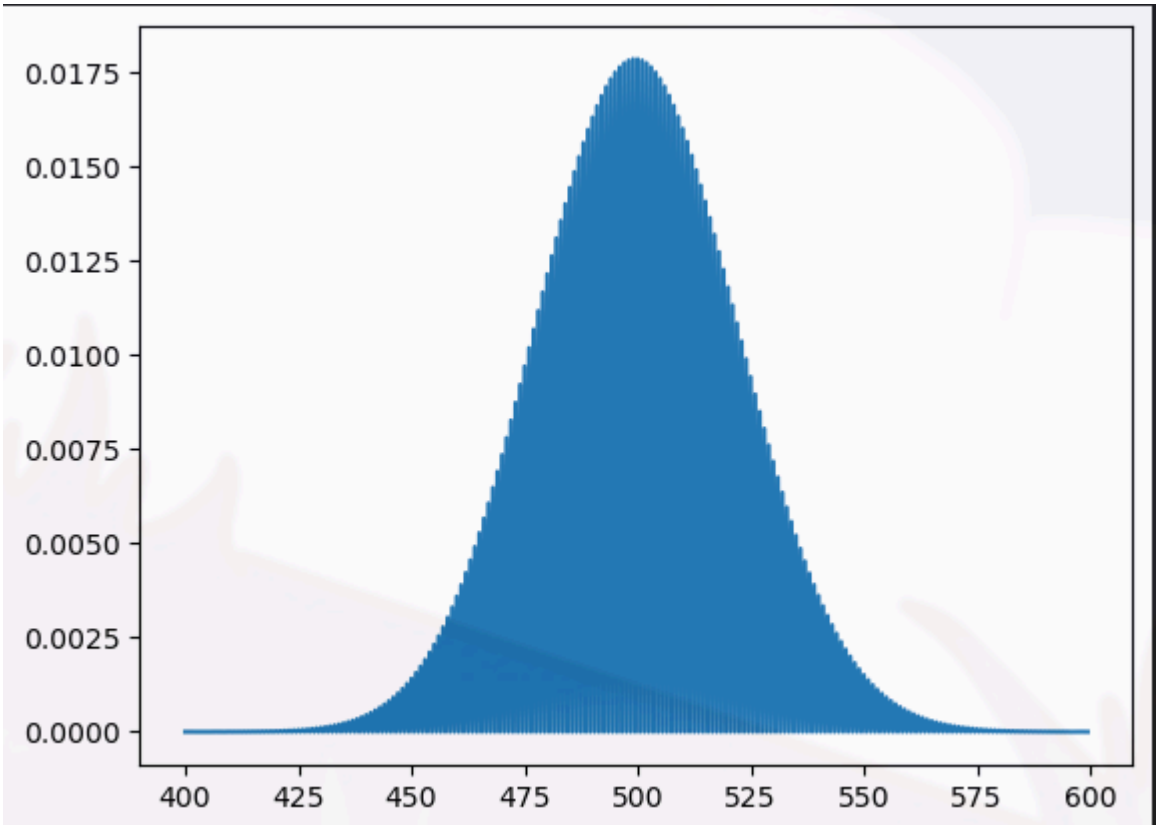
Função de massa de probabilidade binomial, que lida com dados discretos (apenas valores contábeis, ou seja, números inteiros). Ela serve para calcular a chance de sucesso de algo, ou seja, os resultados possíveis são apenas sucesso ou fracasso, nisso podemos testar quantas vezes repetimos os testes, a probabilidade de sucesso em cada caso, e no caso o X dessa função seria a probabilidade de dar certo exatamente aquela quantidade de tentativas, ou seja. No gráfico exemplo:



Dado o número de 10 tentativas, a chance de 5 tentativas de sucesso é de 25%.

Poisson é a função que a gente usa para determinar a chance de um evento acontecer dado a quantidade de vezes que esse evento aconteceu anteriormente em uma situação, podendo ser tempo, lugar ou outra medida. Exemplo: A quantidade de pessoas que visitou um site em uma hora.

Considerando-se que a média de pessoas visitando o site em uma hora é de 500 pessoas.



Essas seriam as chances de ter exatamente 400, 425 450… etc pessoas visitando o site em alguma outra hora qualquer.

Percentil

Percentil é uma indicação a uma posição de um valor em uma lista de valores, ele indica que uma porcentagem dos valores é menor ou igual ao valor indicado. Por exemplo, se o percentil de 50% de uma lista de dados é igual a 2, isso significa que 50% da lista tem valores que são igual ou menor que 2.

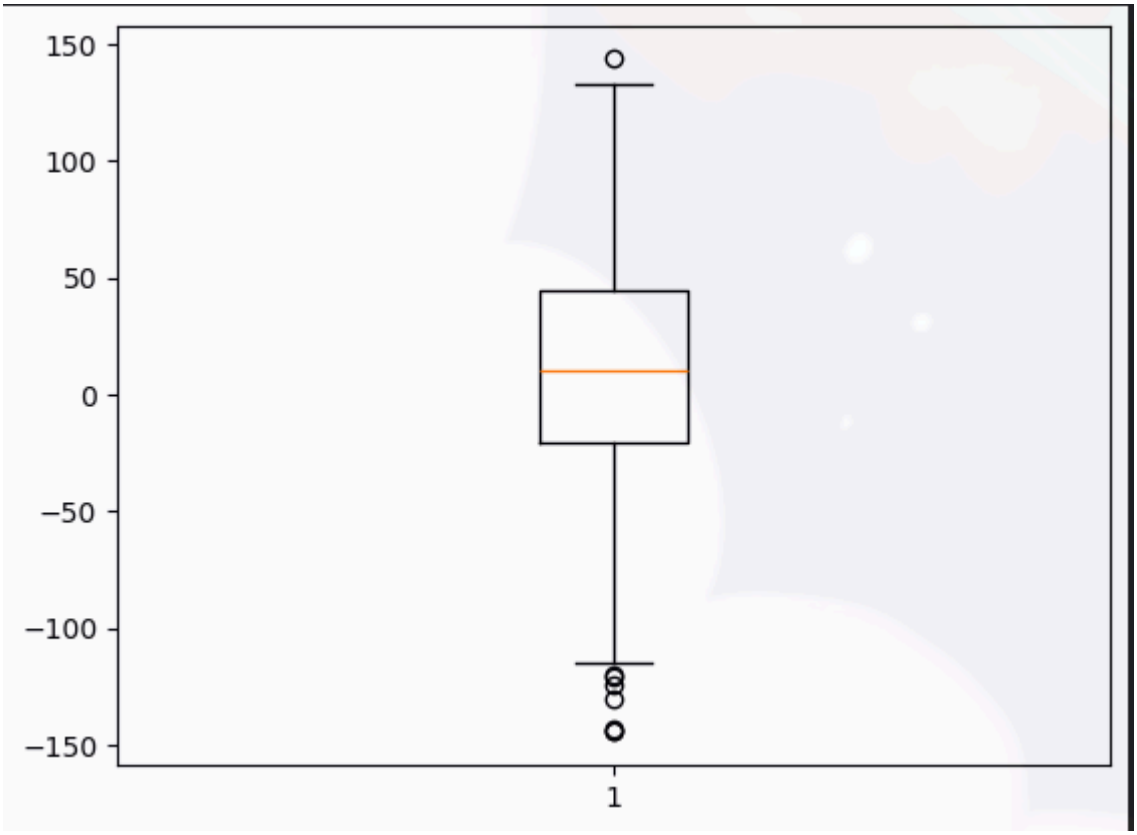
Momentos

São valores que servem para descrever uma função ou uma distribuição. O primeiro momento é a média, o segundo é a variância. O terceiro é skew, em português “assimetria”, serve para ver o quão desequilibrada uma distribuição está em relação a sua média. O quarto momento é “Kurtosis”, em português “curtose”, serve para medir a forma da cauda de uma distribuição de dados, ou seja, quão alta ou fina é essa cauda comparada a uma distribuição normal.

Matplot

Nessa aula foi visto como montar vários tipos de gráficos com a biblioteca do matplot. Nele foi visto como mostrar um gráfico simples, um gráfico com múltiplas linhas, foi ensinado a alterar essas linhas, ajustar os eixos, colocar uma grade de fundo, nomear esses eixos, adicionar legenda, a fazer um gráfico no estilo desenho como se fosse a mão livre, ensinou a fazer gráficos de pizza, gráficos de barras, um gráfico espalhado, um histograma, e por último um Box Whisker Plot, que eu não sei qual seria a tradução.

Esse último acho que é o último que realmente vale a pena explicar, pois é o único que eu não conhecia e achei complexo de entender.



Esse é o gráfico em questão e ele é lido da seguinte forma:

A caixa representa 50% dos dados centrais do conjunto de dados, ou seja, que vai de aproximadamente -40 até 50. A linha dentro da caixa representa onde está a mediana dos dados. Os ‘whiskers’ são a linha que vai até o último valor antes de ser considerado um outlier, ou seja, mostra os valores normais mais afastados. E os outliers são essas bolinhas, que são valores considerados muito afastados da caixa central onde está concentrado os dados.

Seaborn

Depois foi ensinado a montar os gráficos com o seaborn, que permite também montar vários tipos de gráficos diferentes utilizando poucas linhas de comando. Dentre os gráficos que fomos ensinado temos o gráfico de barras, o histograma que tem uma linha de densidade traçado por cima, gráfico de dispersão utilizando o pairplot, que permite ver várias comparações, outro tipo de gráfico de dispersão utilizando scatterplot, outro que contém um histograma junto chamado jointplot, outro que tem uma linha de densidade chamado lmplot. Chegando nos mais complexos tem um grande gráfico de Box Whisker plot, onde é possivel comparar múltiplos dados utilizando o boxplot, depois temos um gráfico que utiliza do swarmplot, que mostra em pontos a dispersão dos dados, o countplot que mostra a quantidade como se fosse um gráfico de barras comuns, e por último um gráfico de calor, utilizando a função heatmap.

Covariância e Correlação

A Covariância mede a variação entre duas variáveis, ou seja, se elas possuem um padrão claro do tipo: uma aumenta enquanto outra diminui, uma aumenta enquanto a outra aumenta, ou não tem nenhum padrão claro de relação entre elas. A correlação serve para normalizar a covariância, conseguindo então verificar se á uma relação perfeita ou se não tem reação baseada nos números -1 0 e 1, onde 0 significa sem relação, onde 1 significa relação perfeita positiva (crescem e diminuem juntas) e -1 significa relação perfeita negativa (relação oposta).

Agora utilizando o python, foi mostrado como calcular esses dois valores e utilizamos de um gráfico de dispersão para comparar com o valor e ver realmente se há realmente uma relação.

Probabilidade Condicional

É a probabilidade de alguma coisa acontecer considerando que outra coisa aconteceu. Por exemplo, se você tem a porcentagem de pessoas que passou no teste 1 e no teste 2, e a porcentagem de pessoas que passou no teste 1, você pode calcular a porcentagem das pessoas que passaram no primeiro teste que também passaram no segundo utilizando a fórmula da probabilidade condicional.

Teoria de Bayes

Serve para atualizar uma probabilidade com uma nova informação. O exemplo utilizado no vídeo foi um utilizadores de droga, se um teste para saber se alguém é usuário de droga tem 99% de acerto, porém 0.3% da população utiliza, então o 99% não seria suficiente.

Prática

No pequeno código prático testando os conhecimentos da aula, eu primeiramente utilizei da biblioteca Faker para criar vários nomes falsos para jogadores de forma aleatória (sem uso prático, apenas estético), depois criei utilizando a distribuição poisson dados aleatórios de média de kills e média de mortes por partida, criando assim um dataframe no pandas com esses dados. Também utilizei do random choice para escolher ranks aleatórios que vão do bronze até mestre para os jogadores.

Comecei fazendo testes simples como média, moda mediana, variância e desvio padrão de uma coluna de dados. Depois criei uma coluna com a quantidade de vitórias de um jogador, essa quantidade de vitórias tem uma relação direta com a quantidade de partidas e com o rank do jogador. E então, usando a biblioteca binomial do numpy coloquei as vitórias para os jogadores de forma relacionada,

Após isso, fiz alguns gráficos de dispersão e um gráfico de barras para vizualizar a relação entre o rank dos jogadores e a quantidade de vitórias. Também usei o gráfico de dispersão para analisar que a quantidade de vitórias não tem nada a ver com a média de kills do jogador, isso porque na hora que eu criei os dados eu não fiz nenhuma relação entre eles.

E por último, fiz alguns pequenos testes para verificar a probabilidade de algumas coisas acontecer considerando o rank e a quantidade de vitórias e partidas de um jogador.

Conclusões

Nesse Card eu pude ver várias funções diferentes em estatísticas e probabilidade. Fui ensinado sobre a melhor ocasião para se utilizar mediana ou quando é melhor utilizar a média. Também foi muito bom entender sobre os diferentes tipos de distribuições de dados e a relação que eles têm com a média e a variância e o desvio padrão.

A parte mais divertida do card foi mexer com os gráficos, pois a visualização facilita muito o entendimento e é muito simples de criar os gráficos utilizando a biblioteca do matplot e do seaborn, com poucas linhas de código é possível ter um bom entendimento de como estão os dados do programa.

Covariância e correlação pra mim foi algo completamente novo também, pois são termos que antes eram desconhecidos, achei bem interessante o conceito e é muito útil para o entendimento da relação entre duas variáveis.

Quanto à probabilidade condicional e a teoria de bayes, que são dois assuntos muito próximos, foram algo um pouco mais complexo de entender, mas não necessariamente mais difícil. A base de entender sobre a chance de algo acontecer e ser diferente se os eventos tiverem relação ou

não também bate completamente com os termos anteriores de covariância e correlação, então achei interessante poder entender isso em poucos minutos de uma videoaula tão bem explicada.

Referências

Videoaulas do curso: Machine Learning, Data Science and Deep Learning with Python na seção 2: Statistics and Probability Refresher, and Python Practice

Uso do faker no código prático:

https://hub.asimov.academy/tutorial/como-gerar-dados-falsos-com-python-utilizando-a-biblioteca-faker/

Para usar a função binomial do random do numpy:

https://numpy.org/doc/2.1/reference/random/generated/numpy.random.binomial.html