Disciplina: Aprendizado Profundo

Aula 1: Introdução, Motivação e Fundamentos

Eliezer de Souza da Silva
sereliezer.github.io
eliezer.silva@ufc.br

Mestrado e Doutorado em Ciência da Computação, Universidade Federal do Ceará (MDCC / UFC)

8 de Setembro de 2025

Roteiro da Aula de Hoje

Oi mundo!

Bem-vindos!

Meu nome é Eliezer. É um prazer tê-los aqui para explorarmos juntos as fronteiras do aprendizado profundo.

- Sou professor visitante no MDCC/UFC e pesquisador do Basque Center for Applied Mathematics (BCAM).

Oi mundo!

Bem-vindos!

Meu nome é Eliezer. É um prazer tê-los aqui para explorarmos juntos as fronteiras do aprendizado profundo.

- Sou professor visitante no MDCC/UFC e pesquisador do Basque Center for Applied Mathematics (BCAM).
- Pesquisa foca em aprendizado de máquina probabilístico, métodos Bayesianos e modelos generativos. Ph.D. em Ciência da Computação pela NTNU, Mestre pela FEEC/Unicamp e Bacharel pela UFES (Eng. Comp.).

Oi mundo!

Bem-vindos!

Meu nome é Eliezer. É um prazer tê-los aqui para explorarmos juntos as fronteiras do aprendizado profundo.

- Sou professor visitante no MDCC/UFC e pesquisador do Basque Center for Applied Mathematics (BCAM).
- Pesquisa foca em aprendizado de máquina probabilístico, métodos Bayesianos e modelos generativos. Ph.D. em Ciência da Computação pela NTNU, Mestre pela FEEC/Unicamp e Bacharel pela UFES (Eng. Comp.).
- Estou animado para conhecer vocês, os seus interesses e contribuir com o aprendizado de todos.

Oi mundo!

Bem-vindos!

Meu nome é Eliezer. É um prazer tê-los aqui para explorarmos juntos as fronteiras do aprendizado profundo.

- Sou professor visitante no MDCC/UFC e pesquisador do Basque Center for Applied Mathematics (BCAM).
- Pesquisa foca em aprendizado de máquina probabilístico, métodos Bayesianos e modelos generativos. Ph.D. em Ciência da Computação pela NTNU, Mestre pela FEEC/Unicamp e Bacharel pela UFES (Eng. Comp.).
- Estou animado para conhecer vocês, os seus interesses e contribuir com o aprendizado de todos.
- Página: `https://sereliezer.github.io/`
- Email: eliezer.silva@ufc.br

O Processo de Aprendizagem

Terence Tao sobre o Aprendizado em Matemática

O ilustre matemático no post "Existe mais do que rigor e provas na matemática" apresenta um *processo cíclico* composto de estágios:

O Processo de Aprendizagem

Terence Tao sobre o Aprendizado em Matemática

O ilustre matemático no post "Existe mais do que rigor e provas na matemática" apresenta um *processo cíclico* composto de estágios:

- **Pré-formal**: Temos intuições, uma atitude lúdica, curiosa e aberta.

O Processo de Aprendizagem

Terence Tao sobre o Aprendizado em Matemática

O ilustre matemático no post "Existe mais do que rigor e provas na matemática" apresenta um *processo cíclico* composto de estágios:

- **Pré-formal**: Temos intuições, uma atitude lúdica, curiosa e aberta.
- **Formal**: Buscamos desenvolver conhecimento rigoroso, de forma paulatina e cautelosa.

O Processo de Aprendizagem

Terence Tao sobre o Aprendizado em Matemática

O ilustre matemático no post "Existe mais do que rigor e provas na matemática" apresenta um *processo cíclico* composto de estágios:

- **Pré-formal**: Temos intuições, uma atitude lúdica, curiosa e aberta.
- **Formal**: Buscamos desenvolver conhecimento rigoroso, de forma paulatina e cautelosa.
- **Pós-formal**: "Esquecemos"os formalismos e reaprendemos a partir dos fundamentos, conectando cada intuição a um formalismo e vice-versa.

Apresentação do Curso I

Estrutura da Disciplina

- **Objetivo:** Construir uma base teórica e prática sólida em redes neurais modernas.
- **Metodologia:** Desenvolver aprendizado em diferentes frentes – Modelagem, Teoria, Técnica e Aplicações. Participação ativa, perspectiva crítica e um "olhar de pesquisa".
- **Comunicação:** Usaremos o SIGAA para notícias, materiais e entregas.

Avaliação e Logística I

Método de Avaliação

O foco é 100% no aprendizado contínuo, integrado e prático. **Não haverá provas.**

- **Listas de Exercícios (60%):** N ($N > 3$) listas teórico-práticas. Alguns laboratórios contarão como listas práticas. Detalhes ainda a definir.
- **Projeto Final (40%):** Mini-projeto de pesquisa em grupos de 3, com duas entregas parciais e um seminário final.

Avaliação e Logística II

Logística Híbrida

- **Presencial:** Setembro, Dezembro e Janeiro.
- **Remoto:** Outubro e Novembro.

Feedback sobre as aulas

Feedback é necessário para aprendizado. Disponibilizarei um link para feedback anônimo, mas sintam-se à vontade para falar abertamente comigo.

Livros-Texto e Leitura Complementar I

Livros-Texto Principais

Estes livros formam a base da nossa disciplina.

- Christopher M. Bishop e Hugh Bishop (2023). *Deep Learning: Foundations and Concepts*. Springer
- Simon J.D. Prince (2023). *Understanding Deep Learning*. The MIT Press
- Kevin Patrick Murphy (2023). *Probabilistic Machine Learning: Advanced Topics*. The MIT Press
- Christopher M. Bishop (2006). *Pattern Recognition and Machine Learning*. Springer

Livros-Texto e Leitura Complementar II

Leitura Complementar Sugerida

Para aprofundar em tópicos específicos que discutiremos.

- Sam Buchanan et al. (ago. de 2025). *Learning Deep Representations of Data Distributions*. https://ma-lab-berkeley.github.io/deep-representation-learning-book/. Online
- Peter D. Grünwald (2007). *The Minimum Description Length Principle*. The MIT Press
- Andreas Krause e Jonas Hübotter (2025). *Probabilistic Artificial Intelligence*. arXiv: 2502.05244 [cs.AI]

O Impacto do Deep Learning no Mundo Real I

Por que estudar Redes Neurais agora?

Estamos vivendo uma revolução tecnológica impulsionada por avanços em arquiteturas de redes neurais.

- **Redes Convolucionais (CNNs):** Dominaram a **Visão Computacional**.
  - *Aplicações:* Carros autônomos, diagnóstico médico por imagem, reconhecimento facial.
- **Transformers:** A arquitetura que define a era da **IA moderna**.
  - *Aplicações:* NLP (*ChatGPT*, *BERT*), tradução, bioinformática (*AlphaFold*).

O Impacto do Deep Learning no Mundo Real II

- **Modelos Generativos (GANs, Difusão):** Criam dados novos e realistas.
  - *Aplicações:* Geração de imagens, vídeos e arte (*Midjourney, Dall-e*), extração de conhecimento.

A Revolução Está Apenas Começando I

O Futuro: Desafios Abertos

Apesar do sucesso, os maiores desafios ainda estão por vir. A próxima geração de avanços irá além das tarefas atuais.

A Revolução Está Apenas Começando II

De Padrões a Sistemas Complexos

Precisamos ir além de "simplesmente" reconhecer padrões para resolver problemas de larga escala na engenharia e na sociedade:

- **Engenharia:** Design autônomo de materiais, controle de redes elétricas inteligentes, sistemas logísticos globais.
- **Ciência:** Modelagem climática, simulação de sistemas biológicos complexos, raciocínio matemático automatizado.
- **Sociedade:** Sistemas de ensino personalizados, otimização de políticas públicas, medicina de precisão.

O Objetivo Deste Curso: De Usuário a Criador I

1. Entender os Princípios de Design

O objetivo é saber **como construir**.

- Escolher a **arquitetura correta**.
- Entender **trade-offs** (custo vs. performance).
- Diagnosticar e **depurar** modelos.
- Criar soluções **inovadoras**.

2. Avaliar Capacidades e Limitações

Desenvolver um olhar crítico para saber **o que é possível**.

- Quando DL é a ferramenta **certa**?
- Reconhecer modos de falha (alucinações).
- Avaliar modelos além da acurácia: robustez, ética e interpretabilidade.

O Objetivo Deste Curso: De Usuário a Criador II

Leitura Adicional

Para uma visão crítica sobre os desafios futuros da IA:

Michael I Jordan (2019). "Artificial intelligence—the revolution hasn't happened yet". Em: *Harvard Data Science Review* 1.1

As Sementes Intelectuais I: Computação e Aprendizado I

Alan Turing (1948): "Intelligent Machinery"

Neste ensaio, Turing foi além da computabilidade e explorou como uma máquina poderia aprender. Turing 1969

- Ele propôs as **"máquinas não-organizadas"** (unorganised machines).
- Eram redes de **circuitos lógicos** simples (e.g., portas NAND) interconectados aleatoriamente.
- A ideia central: um sistema desorganizado poderia, através de "interferência apropriada"(treinamento), se tornar organizado.

As Sementes Intelectuais I: Computação e Aprendizado II

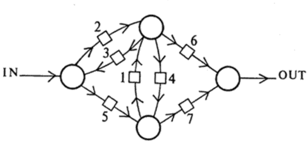

Figura: Ilustração da B-Type Network de Turing, uma precursora das redes neurais.

Leitura Adicional

Para aprofundar sobre as ideias de Turing

Alan M. Turing (1969). "Intelligent Machinery". Em: *Machine Intelligence 5*. Ed. por Bernard Meltzer e Donald Michie. Edinburgh University Press, pp. 3–23

As Sementes Intelectuais II: Conexões Interdisciplinares I

Cibernética

Wiener, McCulloch & Pitts (anos 40-50): A ideia de que sistemas (biológicos ou artificiais) podem aprender e se auto-regular através de *feedback*. Criaram o primeiro modelo matemático do neurônio.

Conexionismo e Ciências Cognitivas

Nos anos 80, o conexionismo ressurgiu como alternativa ao simbolismo, postulando que a inteligência emerge da interação de muitas unidades simples e interconectadas.

O "Teseu" de Shannon (1950), um rato artificial que aprendia a navegar em um labirinto.

As Sementes Intelectuais II: Conexões Interdisciplinares II

Leitura Adicional

Artigo trazendo uma narrativa detalhada da vida e obra de Pitts e sua relação com McCulloch e outros intelectuais da sua época

Amanda Gefter (fev. de 2015). "The Man Who Tried to Redeem the World with Logic". Em: *Nautilus*. URL: https://nautil.us/the-man-who-tried-to-redeem-the-world-with-logic-235253/

Teoria da Informação

Claude Shannon (1948): Formalizou o conceito de *informação* e entropia. Nossos modelos aprendem para extrair informação dos dados e reduzir a incerteza sobre uma previsão.

As Sementes Intelectuais II: Conexões Interdisciplinares III

Leitura Adicional

Artigo seminal que fundamenta a área de Teoria de Informação

Claude E. Shannon (1948). *A Mathematical Theory of Communication*. Vol. 27, pp. 379–423, 623–656

Visão Panorâmica dos Fundamentos

O Que Vamos Explorar

Nessa seção, iremos explorar de forma conceitual os fundamentos matemáticos necessários, construindo uma intuição sólida.

- Teoria do Aprendizado Estatístico
- Teoria da Probabilidade
- Teoria da Informação
- Teoria da Otimização (próximas aulas)

Fundamento 1: Teoria do Aprendizado Estatístico I

O Espaço de Hipóteses e o Risco

- O espaço de funções que exploramos é o **espaço de hipóteses**, $\mathcal{H}$.
- O aprendizado consiste em escolher uma hipótese $h \in \mathcal{H}$ que generalize bem para dados não vistos.
- **Risco Empírico ($\hat{R}$):** O erro que medimos no conjunto de treino.
- **Risco Real (R):** O erro esperado na distribuição real dos dados (o que realmente queremos minimizar).

Fundamento 1: Teoria do Aprendizado Estatístico II

Espaços de Funções/Hipóteses ($\mathcal{H}$)

Onde procuramos por h? Em uma classe de modelos (ou espaço de funções) $\mathcal{H}$.

- **Regressão Linear:** $\mathcal{H}$ é o espaço de funções lineares $h(x) = w^\top x$.
- **Redes Neurais:** $\mathcal{H}$ é um espaço vasto e altamente expressivo de funções não-lineares.

O "aprendizado" é um processo de **busca** pela função $h^* \in \mathcal{H}$ que satisfaça alguns critérios.

Fundamento 1: Teoria do Aprendizado Estatístico III

O Dilema Central

Queremos minimizar o Risco Real, mas só podemos calcular o Risco Empírico. A diferença entre eles é o **gap de generalização**, e seu tamanho depende da complexidade do espaço de hipóteses $\mathcal{H}$.

Mergulho Raso: Minimum Description Length (MDL) I

O Princípio MDL: Aprendizado como Compressão

A melhor hipótese (modelo) para explicar um conjunto de dados é aquela que permite a **compressão máxima** dos dados, resultando na descrição mais curta possível.

Mergulho Raso: Minimum Description Length (MDL) II

O Código de Duas Partes e a Navalha de Occam

O comprimento total da descrição dos dados D é a soma do comprimento da descrição da hipótese H e do comprimento da descrição dos dados codificados com a ajuda de H:

$$L(D) \approx \underbrace{L(H)}_{\text{Complexidade do Modelo}} + \underbrace{L(D|H)}_{\text{Ajuste aos Dados (Erro)}}$$

A complexidade $L(H)$ pode ser entendida como o **tamanho do programa** que implementa o modelo. A Indução de Solomonoff formaliza isso, postulando que a melhor previsão é uma média de todos os programas que geram os dados, com peso maior para os programas **mais curtos**.

Mergulho Raso: Minimum Description Length (MDL) III

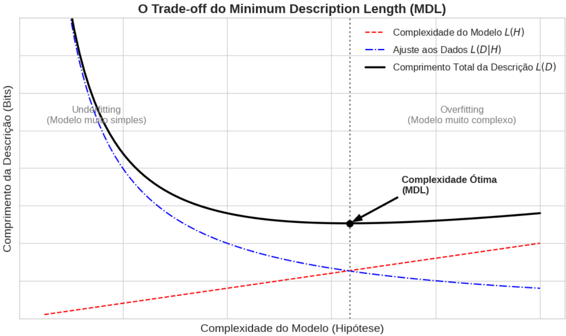

Figura: MDL busca o balanço: um modelo muito simples não captura os dados; um modelo muito complexo custa caro para ser descrito (um "programa"longo) e não generaliza.

Fundamento 2: A Linguagem da Incerteza - Probabilidade I

Axiomas da Probabilidade (Kolmogorov)

A teoria da probabilidade é uma extensão da lógica, construída sobre 3 axiomas:

1. A probabilidade de um evento é não-negativa: $P(A) \geq 0$.
2. A probabilidade do espaço amostral (evento certo) é 1: $P(\Omega) = 1$.
3. A probabilidade da união de eventos disjuntos é a soma de suas probabilidades.

A partir destes, derivamos as regras de soma e produto.

Fundamento 2: A Linguagem da Incerteza - Probabilidade II

Modelos Gráficos e Independência

- Permitem representar distribuições complexas sobre muitas variáveis de forma compacta, codificando suposições de **independência condicional**.
- Exemplo (Rede Bayesiana): $A \to B \to C$. A distribuição conjunta fatoriza:

$$p(A, B, C) = p(A)p(B|A)p(C|B)$$

- Redes neurais profundas podem ser vistas como um tipo de modelo gráfico, onde as camadas representam variáveis latentes.

Regras Fundamentais da Probabilidade I

Regra do Produto e da Soma

A partir dos axiomas, derivamos duas regras essenciais:

- **Regra da Soma:** $p(X) = \sum_Y p(X, Y)$ (Marginalização)
- **Regra do Produto:** $p(X, Y) = p(Y|X)p(X) = p(X|Y)p(Y)$

Regras Fundamentais da Probabilidade II

A Regra de Bayes

Combinando as duas formas da regra do produto, obtemos a Regra de Bayes, fundamental para o aprendizado:

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}$$

$$\text{Posterior} = \frac{\text{Verossimilhança} \times \text{Prior}}{\text{Evidência}}$$

Ela nos permite "inverter"a inferência: a partir do que observamos (X), atualizamos nossa crença sobre o que não vemos (Y).

Distribuições Típicas e Momentos I

Momentos de uma Distribuição

Momentos descrevem a forma de uma distribuição. Os mais importantes são:

- **Média (1º Momento):** O valor esperado. $\mathbb{E}[X] = \int xp(x)dx$
- **Variância (2º Momento Central):** A dispersão em torno da média. $\mathrm{var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2]$

Distribuições Típicas e Momentos II

Distribuições de Probabilidade Comuns

- **Gaussiana (Normal):** Usada para modelar quantidades contínuas (e.g., erros em regressão).

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- **Bernoulli:** Usada para modelar um evento binário (e.g., classificação com duas classes).

$$\text{Bern}(x|\mu) = \mu^x(1-\mu)^{1-x} \quad \text{para } x \in \{0, 1\}$$

O Princípio da Máxima Verossimilhança (MLE) I

Como encontrar os melhores pesos w? Escolhemos aqueles que tornam os dados observados $\mathcal{D}$ os mais prováveis.

A Função de Verossimilhança (Likelihood)

Assumindo que os dados são i.i.d., a probabilidade de todo o conjunto de dados de alvos $\mathrm{y} = \{y_1, \ldots, y_N\}$ é:

$$p(\mathrm{y}|\mathrm{X}, \mathrm{w}) = \prod_{n=1}^{N} p(y_n|\mathrm{x}_n, \mathrm{w})$$

O objetivo do **Maximum Likelihood Estimation (MLE)** é encontrar os pesos w_{ML} que maximizam esta função C. M. Bishop e H. Bishop 2023.

O Princípio da Máxima Verossimilhança (MLE) II

Log-Likelihood

Na prática, maximizamos o logaritmo da verossimilhança, que transforma o produto em uma soma e é numericamente mais estável:

$$\ln p(\mathrm{y}|\mathrm{X}, \mathrm{w}) = \sum_{n=1}^{N} \ln p(y_n|\mathrm{x}_n, \mathrm{w})$$

De Máxima Verossimilhança a Funções de Custo I

Maximizar o log-likelihood é equivalente a **minimizar o negativo do log-likelihood (NLL)**.

Regressão (Likelihood Gaussiano) → Erro Quadrático

Assumindo que o alvo y_n é a saída da rede $g(\mathrm{x}_n, \mathrm{w})$ mais um ruído Gaussiano, temos $p(y_n|\mathrm{x}_n, \mathrm{w}) = \mathcal{N}(y_n|g(\mathrm{x}_n, \mathrm{w}), \sigma^2)$. O NLL do dataset é:

$$-\ln p(\mathrm{y}|\mathrm{X}, \mathrm{w}) = \frac{1}{2\sigma^2}\sum_{n=1}^{N}(y_n - g(\mathrm{x}_n, \mathrm{w}))^2 + \frac{N}{2}\ln(2\pi\sigma^2)$$

Minimizar isso em relação a w é o mesmo que minimizar o **erro da soma dos quadrados (sum-of-squares error)**.

De Máxima Verossimilhança a Funções de Custo II

Classificação (Likelihood de Bernoulli) → Cross-Entropy

Para um alvo binário $y_n \in \{0, 1\}$, assumimos $p(y_n|\mathrm{x}_n, \mathrm{w}) = \mathrm{Bern}(y_n|g(\mathrm{x}_n, \mathrm{w}))$. O NLL é:

$$-\ln p(\mathrm{y}|\mathrm{X}, \mathrm{w}) = -\sum_{n=1}^{N} \{y_n \ln g(\mathrm{x}_n, \mathrm{w}) + (1 - y_n) \ln(1 - g(\mathrm{x}_n, \mathrm{w}))\}$$

Esta é exatamente a função de custo de **entropia cruzada binária (binary cross-entropy)**.

Fundamento 3: Teoria da Informação I

Mergulho Raso: De Onde Vem o Logaritmo na Entropia?

Queremos uma função $S(p)$ que meça a "surpresa"de um evento com probabilidade p. Intuitivamente, ela deve satisfazer:

1. Surpresa é não-negativa e decresce com p.
2. A surpresa de dois eventos **independentes** ocorrendo juntos é a **soma** de suas surpresas individuais:

$$S(p_1 \cdot p_2) = S(p_1) + S(p_2)$$

A única família de funções que satisfaz a propriedade aditiva (2) é a logarítmica, $S(p) = -C \log(p)$. A Entropia é, portanto, a surpresa média de uma distribuição, $\mathbb{E}[S(p(X))]$.

Fundamento 3: Teoria da Informação II

Entropia

Para uma variável aleatória X com função de probabilidade p, a entropia mede a "surpresa" média Claude E. Shannon 1948:

$$H(p) = -\sum_{k=1}^{K} p(x_k) \log_2 p(x_k)$$

É a quantidade média de informação (em bits) que ganhamos ao observar uma amostra de X.

- Uma moeda honesta ($P(\text{cara}) = 0.5$) tem entropia máxima (1 bit). Cada resultado é igualmente surpreendente.
- Uma moeda viciada ($P(\text{cara}) = 0.99$) tem entropia próxima de zero, então não há surpresa.

Fundamento 3: Teoria da Informação III

Divergência KL e Cross-Entropy

A **Divergência Kullback-Leibler (KL)** mede a dissimilaridade entre duas distribuições p e q:

$$\mathrm{KL}(P||Q) = -\sum_{k=1}^{K} p(x_k) \log_2 \left(\frac{q(x_k)}{p(x_k)} \right) \geq 0$$

Isso nos leva à **Cross-Entropy**, $H(p, q)$, que é a base para a função de custo em classificação. Minimizar a cross-entropy é equivalente a minimizar a divergência KL entre a distribuição real dos dados (P) e a previsão do nosso modelo (Q) C. M. Bishop 2006.

$$H(P, Q) = H(P) + \mathrm{KL}(P||Q)$$

Anatomia de uma Rede Neural I

Blocos de Construção

Uma rede neural é um **grafo computacional** composto em camadas, onde cada neurônio (ou unidade) realiza uma operação simples:

- **Transformação Linear:** Uma soma ponderada das entradas, mais um viés. $z = \mathrm{w}^T \mathrm{x} + b$
- **Função de Ativação:** Uma transformação não-linear aplicada ao resultado. $a = \sigma(z)$

Os **pesos** (w) e **vieses** (b) são os parâmetros que o modelo aprende durante o treinamento.

Anatomia de uma Rede Neural II

Composição em Camadas

A saída de uma camada de neurônios, $a^{(l)} = \sigma(W^{(l)} a^{(l-1)} + b^{(l)})$, serve como entrada para a próxima. Essa composição hierárquica é o que permite que redes profundas aprendam representações (features) cada vez mais complexas dos dados.

Aproximação de Funções com Funções de Base I

A Ideia Central da Teoria da Aproximação

Qualquer função "bem comportada" pode ser representada como uma **soma ponderada** de um conjunto de funções mais simples (as "funções de base" ϕ_i).

$$g(x) = \sum_{i=1}^{M} w_i \phi_i(x)$$

Exemplos clássicos de funções de base:

- **Polinômios:** Base para a expansão em Série de Taylor.
- **Senos e Cossenos:** Base para a representação em Série de Fourier.

Aproximação de Funções com Funções de Base II

- **Funções Radiais de Base (RBFs):** Usadas em SVMs e outros métodos.

Conexão com Redes Neurais

Uma camada de uma rede neural calcula uma combinação linear (soma ponderada) das saídas da camada anterior. Essas saídas atuam como **funções de base não-lineares que são aprendidas** a partir dos dados, em vez de serem fixas. Portanto a classe de modelos com funções de base $g(x) = \sum_{i=1}^{M} w_i \phi_i(x)$ define uma **rede neural rasa**, ou seja, de apenas uma ou poucas camadas.

Exemplo Prático: Rede Rasa vs. Aprendizado de Features I

Para ilustrar a diferença entre bases fixas e bases aprendidas, vamos analisar um exemplo prático.

Laboratório Interativo no Google Colab

O código completo para este exemplo está disponível como um notebook interativo. Recomendo que todos abram e executem o código para explorar os resultados.

Abrir Notebook no Colab

Exemplo Prático: Rede Rasa vs. Aprendizado de Features II

Objetivo do Experimento

Vamos tentar aproximar um polinômio desconhecido usando uma rede RBF de duas maneiras:

1. **Modelo 1 (Raso):** Centros e larguras das RBFs são fixos. Apenas os pesos da camada final são aprendidos (via Mínimos Quadrados).
2. **Modelo 2 (Deep-like):** Centros, larguras e pesos são todos aprendidos simultaneamente via Descida de Gradiente.

Resultados do Experimento

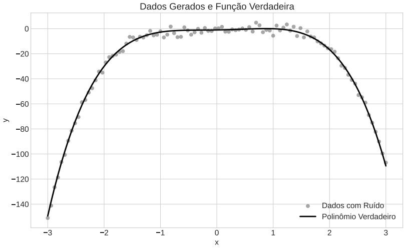

Figura: Passo 1: Geramos 100 pontos de dados com ruído a partir de um polinômio aleatório (a "verdade"que queremos descobrir).

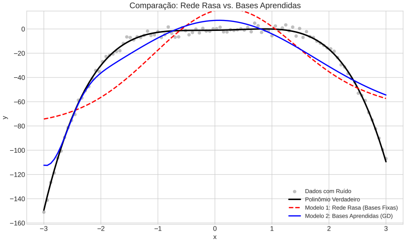

Figura: Passo 2: O modelo de bases aprendidas (azul) se ajusta muito melhor à função verdadeira do que o modelo de bases fixas (vermelho).

Conclusão do Exemplo: A Formulação

A rede RBF implementada tem a seguinte forma:

$$g(x) = \sum_{i=1}^{M} w_i \phi_i(x) + w_0 \text{ onde } \phi_i(x) = \exp\left(-\frac{(x - c_i)^2}{2\sigma_i^2}\right)$$

Modelo 1

Os parâmetros das funções de base (c_i, σ_i) são **fixos**. O aprendizado se resume a encontrar os pesos w que resolvem um problema de Mínimos Quadrados. A não-linearidade é pré-definida, não aprendida.

Modelo 2

Os parâmetros das funções de base (c_i, σ_i) **também são aprendidos** via otimização. A rede descobre as representações (ou features) mais eficientes diretamente a partir dos dados.

O Mapa do Nosso Curso

Nossa Jornada

Vamos explorar diferentes formas de construir o espaço de hipóteses $\mathcal{H}$:

1. Começaremos com o **Perceptron**, a unidade mais básica.
2. Construiremos **FNNs** para aproximação de funções gerais.
3. Especializaremos para diferentes tipos de dados:
   - Sequências com **RNNs** e **Transformers**.
   - Grafos com **GNNs**.
4. Exploraremos modelos que aprendem a própria distribuição dos dados $p(x)$ para gerar amostras novas (**Modelos Generativos**).

Encerramento e Perguntas

Para a Próxima Aula (Quarta-feira)

- **Tópico:** O Perceptron e a Descida de Gradiente.
- **Objetivo:** Implementar nosso primeiro algoritmo de aprendizado.

Tarefa

- Comecem a pensar e a formar os **grupos de 3 pessoas** para o projeto final.

Perguntas?

Referências I

- Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Bishop, Christopher M. e Hugh Bishop (2023). *Deep Learning: Foundations and Concepts*. Springer.
- Buchanan, Sam et al. (ago. de 2025). *Learning Deep Representations of Data Distributions*. https://ma-lab-berkeley.github.io/deep-representation-learning-book/. Online.
- Gefter, Amanda (fev. de 2015). "The Man Who Tried to Redeem the World with Logic". Em: *Nautilus*. URL: https://nautil.us/the-man-who-tried-to-redeem-the-world-with-logic-235253/.
- Grünwald, Peter D. (2007). *The Minimum Description Length Principle*. The MIT Press.

Referências II

- Jordan, Michael I (2019). "Artificial intelligence—the revolution hasn't happened yet". Em: *Harvard Data Science Review* 1.1.
- Krause, Andreas e Jonas Hübotter (2025). *Probabilistic Artificial Intelligence*. arXiv: 2502.05244 [cs.AI].
- Minsky, Marvin e Seymour A Papert (1969). *Perceptrons: An introduction to computational geometry*. MIT press.
- Murphy, Kevin Patrick (2023). *Probabilistic Machine Learning: Advanced Topics*. The MIT Press.
- Prince, Simon J.D. (2023). *Understanding Deep Learning*. The MIT Press.
- Rosenblatt, Frank (1958). "The perceptron: a probabilistic model for information storage and organization in the brain.". Em: *Psychological review* 65.6, p. 386.

Referências III

- Shannon, C. E. (1988). "Programming a computer for playing chess". Em: *Computer Chess Compendium*. Berlin, Heidelberg: Springer-Verlag, pp. 2–13. ISBN: 0387913319.
- Shannon, Claude E. (1948). *A Mathematical Theory of Communication*. Vol. 27, pp. 379–423, 623–656.
- Turing, Alan M. (1969). "Intelligent Machinery". Em: *Machine Intelligence 5*. Ed. por Bernard Meltzer e Donald Michie. Edinburgh University Press, pp. 3–23.