

# Meta-aprendizado

1001513 – Aprendizado de Máquina 2  
Turma A – 2022/2  
Prof. Murilo Naldi



[naldi@ufscar.br](mailto:naldi@ufscar.br)



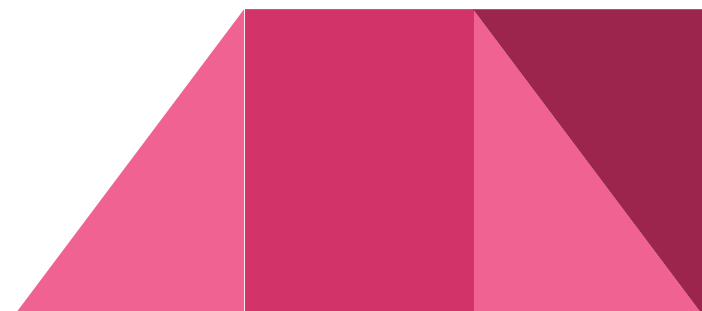
# Agradecimentos

- Pessoas que colaboraram com a produção deste material: Diego Silva
- Intel IA Academy

# Problema

Algoritmos de aprendizado de máquina necessitam de dados

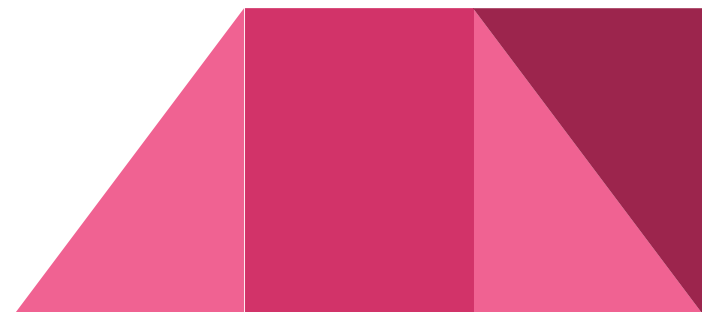
- Mas nem sempre existe uma quantidade grande de dados para uma aplicação específica
- Nesses casos, conhecimento do domínio poderia ajudar
  - Talvez extraído de exemplos semelhantes, mas não iguais
  - Talvez esse conhecimento esteja refletido em modelos já treinados



# Problema

Adicionalmente, sabemos que:

- Diferentes algoritmos podem obter resultados distintos
  - Para diferentes tipos de dados
- Algoritmos possuem hiper-parâmetros a serem definidos
  - Muitas vezes são difíceis de estimar
    - Exemplo:  $k$ , limiares, etc...



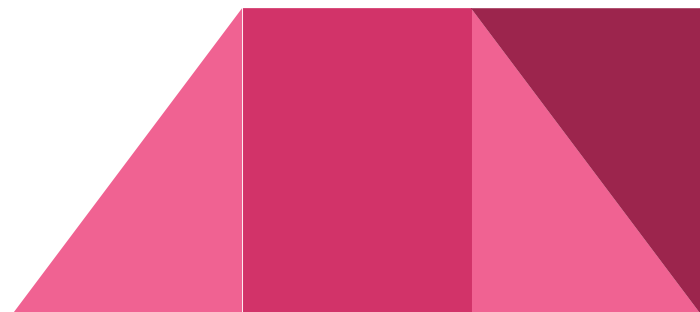
# Meta-aprendizado



# No free lunch

Lembrando que (ainda) não há um algoritmo de AM melhor que os outros para todos os cenários

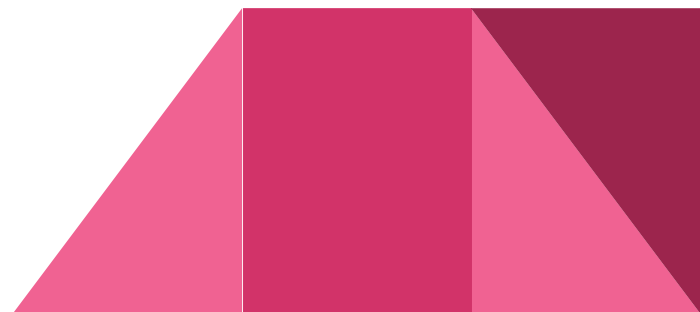
- Depende de **como seu viés indutivo se adapta ao problema**
  - **Problema** = características dos dados



# Meta-aprendizado

O meta-aprendizado foca em resultados anteriores de aprendizado

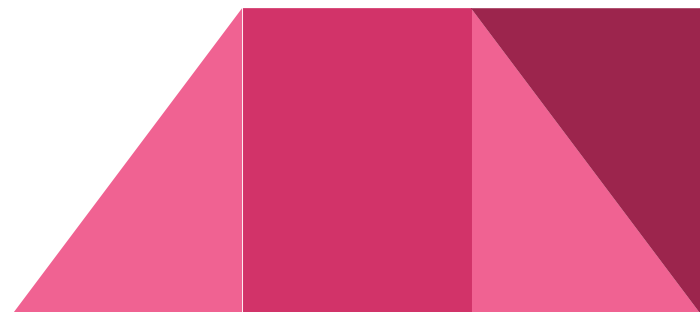
- Por meio de resultados obtidos em **outros conjuntos** de dados
  - Experiência E (definição de Mitchell)
- Aprender a **relação entre problemas e técnicas** de AM



# Meta-aprendizado

O meta-aprendizado foca em resultados anteriores de aprendizado

- Pode ser utilizado para:
  - Encontrar o algoritmo apropriado a um problema específico é conhecido como *Algorithm Selection Problem (ASP)*
  - Encontrar os melhores (hiper-)parâmetros para um algoritmo é conhecido como *Hyper-Parameter Optimization (HPO)*
  - Outros...

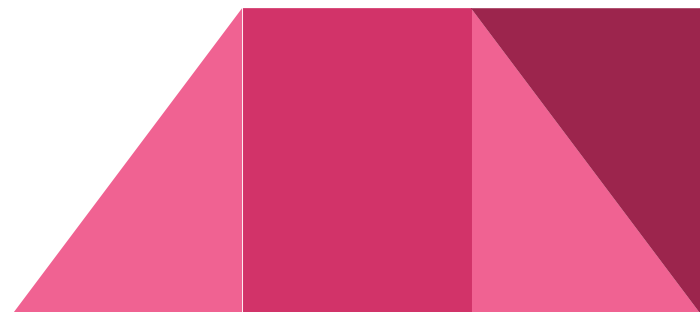




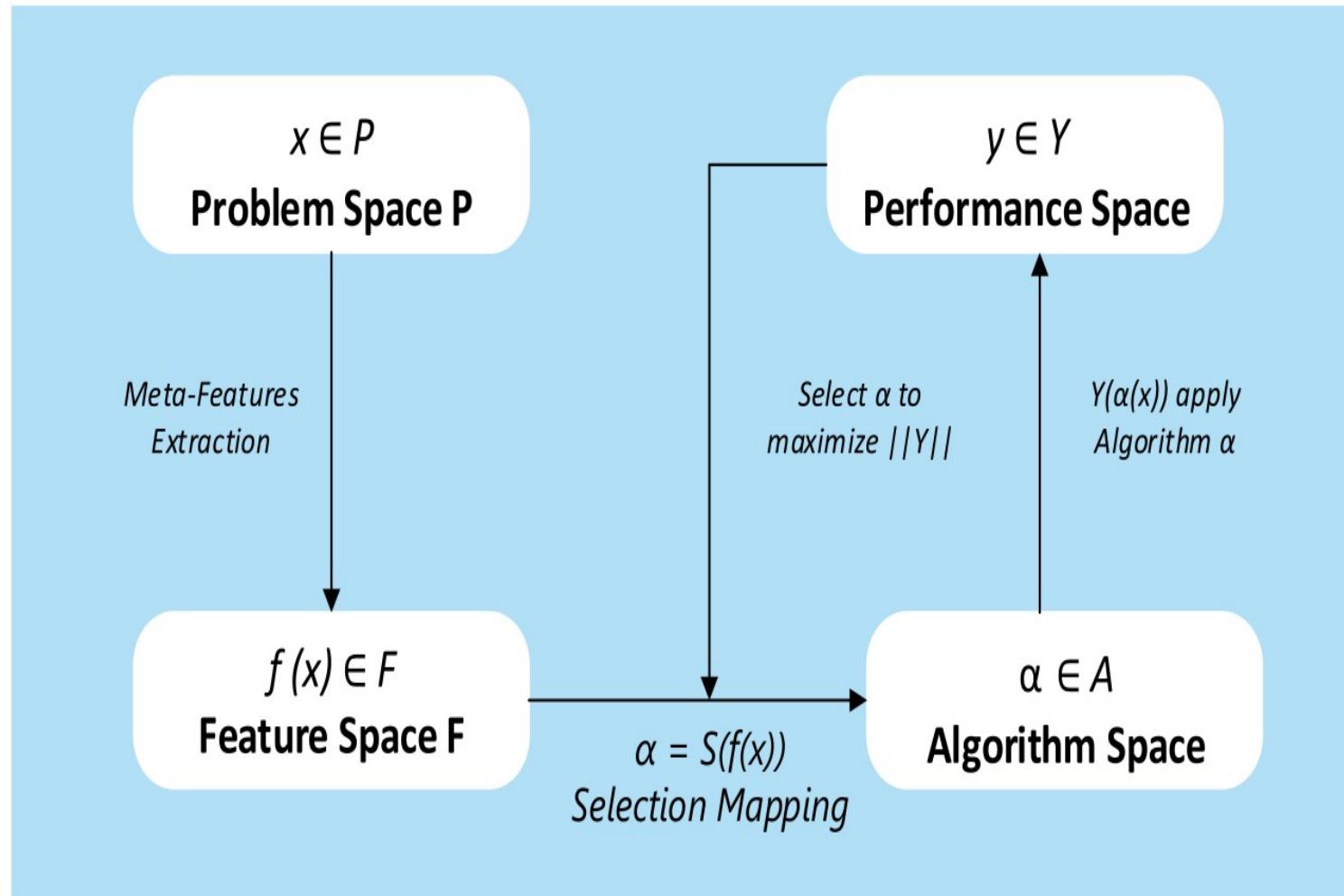
# Definições

Na recomendação de algoritmos

- Input do treino: Um conjunto de conjuntos de dados (**meta-exemplos**) com o algoritmo que teve melhor desempenho em cada um (**meta-alvos**)
  - Os **meta-atributos** são as descrições desses dados (já explico)
- O modelo gerado após a etapa de **meta-treinamento** é chamado **meta-modelo**



# Modelo de Rice para *ASP*

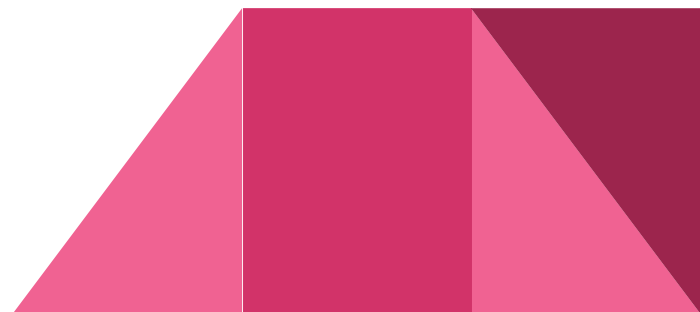


Adaptado de K. A. Smith-Miles, "Cross-disciplinary perspectives on meta-learning for algorithm selection," ACM Comput. Surv., vol. 41, no. 1, pp. 1–25, Dec. 2008. para J. R. Rice, "The algorithm selection problem," Adv. Comput., vol. 15, pp. 65–118, Jan. 1976.

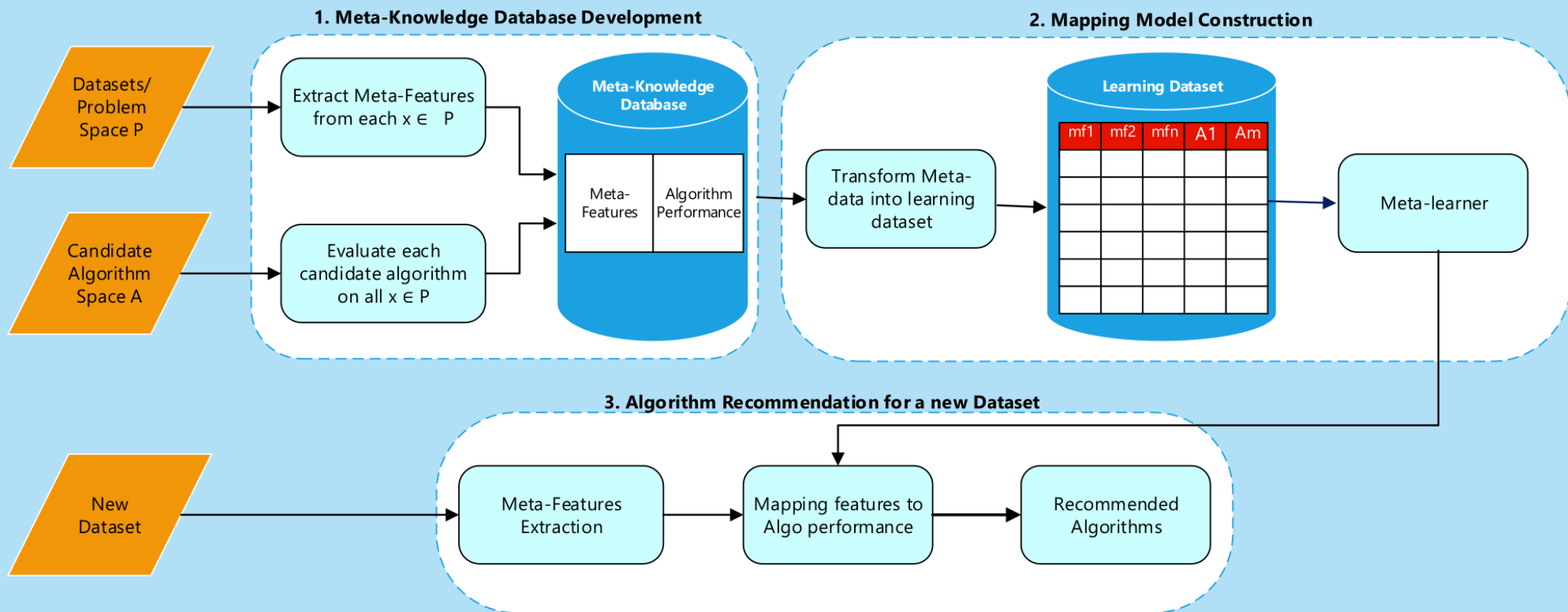
# Problema modelo Rice

Alta complexidade do espaço de problemas  $P$  e de algoritmos  $A$

Meta-aprendizagem ataca essa questão escolhendo problemas de complexidade variável, ao mesmo tempo em que seleciona algoritmos com vieses distintos



# Arcabouço para recomendação de algoritmo



I. Khan, X. Zhang, M. Rehman and R. Ali, "A Literature Survey and Empirical Study of Meta-Learning for Classifier Selection," in IEEE Access, vol. 8, pp. 10262-10281, 2020, doi: 10.1109/ACCESS.2020.2964726.

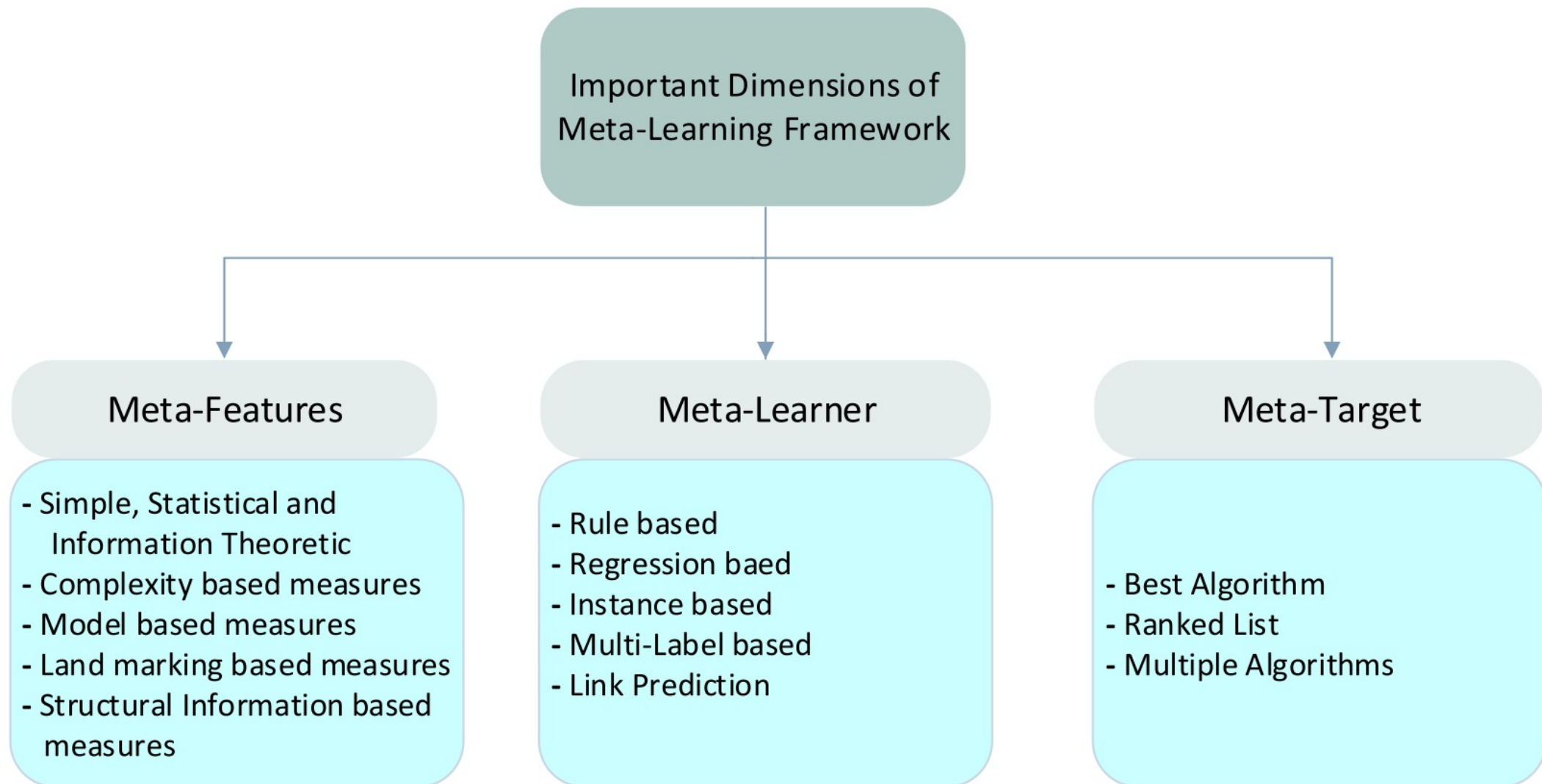
# O que é importante para o meta-aprendizado?

Para um arcabouço de meta-aprendizado é preciso considerar:

- Como/quais serão os meta-atributos?
- Quais são os algoritmos que irão aprender a aprender?
  - *Meta-learners*
- Qual é o alvo?
  - *Meta-target*



# O que é importante para o meta-aprendizado?

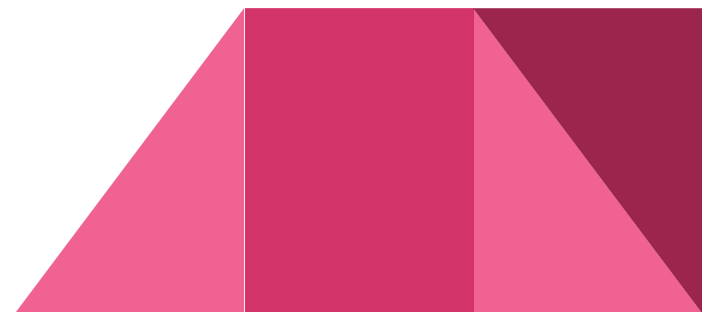


# Caracterização dos dados

Meta-aprendizado pode ser visto como extrair as características que descrevem os dados e usá-las para aprender novos dados de forma mais eficiente

Pode ser feito de forma:

- direta
  - estatística e estrutural
  - complexidade
  - estrutural
- propriedades de modelos
- *landmarking*

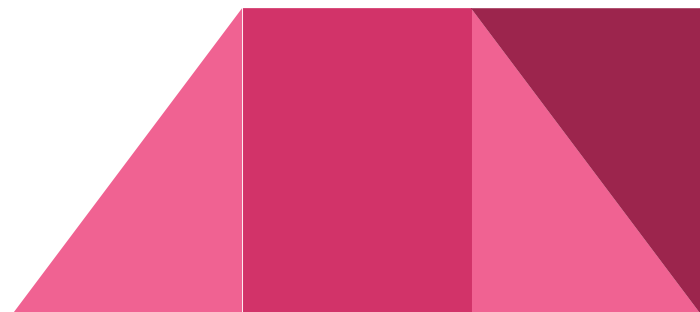


# Caracterização dos dados

## Caracterização direta

- Simples
  - Número de exemplos, classes e atributos, número de atributos categóricos e numéricos, ...
- Estatísticos
  - Média, desvio padrão, obliquidade, distribuição ...
- Teoria da informação
  - Entropia, informação mútua, auto-valores ...

Sugestão: olhar o <https://pypi.org/project/pymfe/>



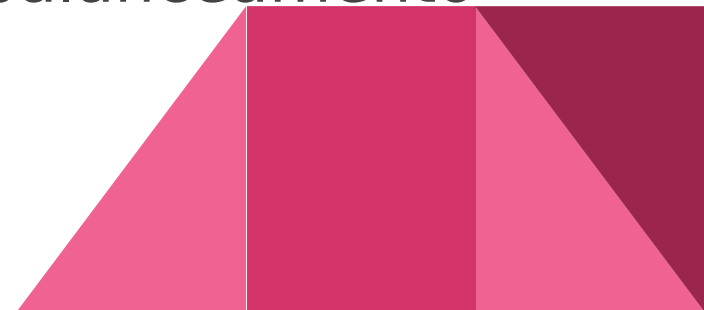


# Caracterização dos dados

## Caracterização direta

- Baseada em complexidade
  - Valor máximo do discriminante de Fisher,
  - Volume de região de sobreposição, linearidade
  - Baseado m vizinhança, fração de pontos de borda
  - Dimensionalidade (com ou sem PCA)
  - Entropia proporção de classes, desbalanceamento

Sugestão: olhar o <https://pypi.org/project/pymfe/>



# Caracterização dos dados

## Caracterização baseada em modelo

- A ideia é induzir modelos e caracterizar o conjunto de dados por características dos modelos induzidos
- Por exemplo, se induzirmos uma Árvore de Decisão
  - Número de folhas, profundidade, entropia média nas folhas...
- Também é possível estudar quantas vezes um atributo é escolhido para decisão

Sugestão: olhar o <https://pypi.org/project/pymfe/>

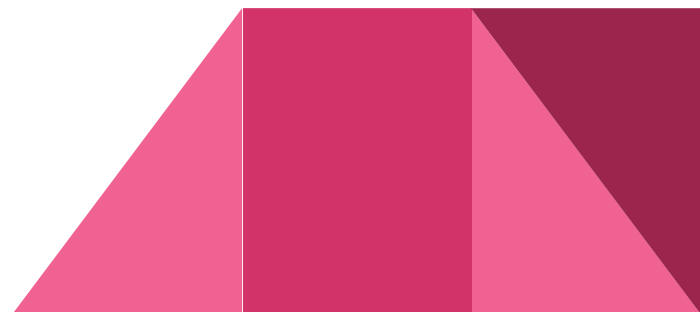


# Caracterização dos dados

## Caracterização baseada em *landmarking*

- Os conjuntos de dados são caracterizados pelo desempenho de diferentes algoritmos de AM aplicados a eles
  - Acurácia, AUC, *recall*...

Sugestão: olhar o <https://pypi.org/project/pymfe/>



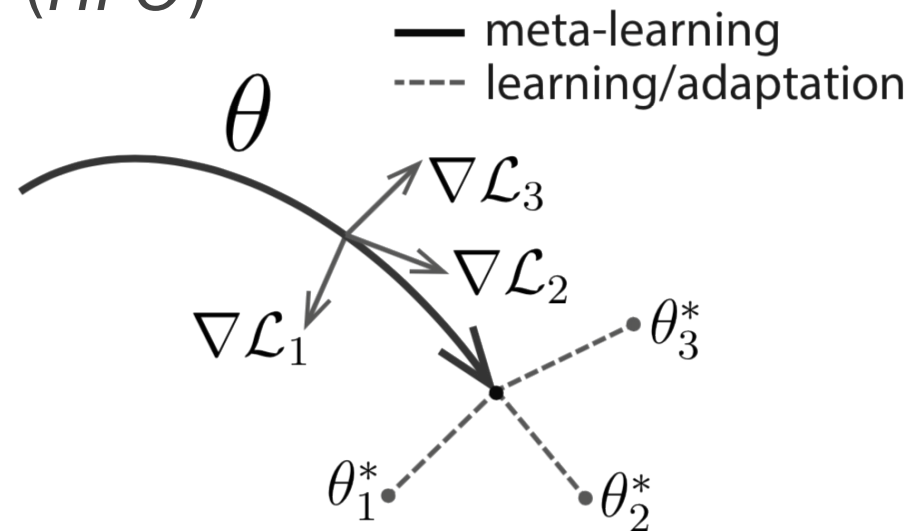
# Escolha do *meta-learner* (Zhang et. al 2020)

RQ1		RQ2	RQ3	RQ4					RQ5	RQ6
Ref	Datasets	Criteria	Algo	Meta Features					Meta learner	Meta target
				SI	PC	MS	LM	STI		
[72]	77	Accuracy	3	✓					KNN	Ranked List
[73]	47	Accuracy	10	✓		✓	✓		KNN	Ranked List
[71]	53	Accuracy+Run Time	10	✓			✓		KNN	Ranked List
[68]	100	Accuracy+Run Time	8	✓					C5.0(Rule Based)	Multiple Algorithms
[13]	84	Accuracy+Run Time	13	✓	✓	✓	✓	✓	ML-KNN	Multiple Algorithms
[66]	84	Accuracy+Run Time	17	✓	✓	✓	✓	✓	KNN	Ranked List
[46]	115	Accuracy+Run Time	22	✓	✓	✓	✓	✓	KNN	Ranked List
[6]	84	Accuracy+Run Time	17	✓	✓	✓	✓	✓	Cluster	Multiple Algorithms
[12]	131	Accuracy+Run Time	21	✓	✓	✓	✓	✓	Link Pred	Ranked List
[51]	39	Accuracy+Run Time	18	✓			✓		KNN	Ranked List
[67]	80	Accuracy+Run Time	11	✓	✓	✓	✓	✓	KNN	Ranked List
[18]	40	Accuracy+Run Time	6	✓	✓				Regression	Best Algorithm
[39]	90	Accuracy+Run Time	5	✓	✓	✓	✓		KNN	Ranked List
[57]	53	Accuracy	6	✓	✓	✓	✓		Regression	Best Algorithm
[77]	85	Accuracy	15	✓			✓		Cluster	Multiple Algorithms
[74]	54	Accuracy	9	✓		✓	✓		Regression	Best Algorithm
[75]	65	Accuracy	8	✓			✓		Regression	Best Algorithm
[69]	67	Accuracy+Run Time	10	✓		✓	✓		KNN	Ranked List
[53]	22	Accuracy	22	✓					C4.5(Rule Based)	Multiple Algorithms
[76]	57	Accuracy	6	✓					Cluster	Multiple Algorithms

# Otimização com meta-aprendizado

Etapas/técnicas baseadas em otimização

- *Hyper-Parameter Optimization (HPO)*
- Redes neurais
  - Reduzir *loss function*  $L$
  - Vetor de parâmetros  $\theta$



Exemplos: Model-Agnostic Meta-Learning (MAML) , Reptile, etc... <https://arxiv.org/abs/1703.03400>

# Model-Agnostic Meta-Learning (Finn et.al 2017)

---

## Algorithm 1 Model-Agnostic Meta-Learning

---

**Require:**  $p(\mathcal{T})$ : distribution over tasks

**Require:**  $\alpha, \beta$ : step size hyperparameters

1: randomly initialize  $\theta$

2: **while** not done **do**

3:     Sample batch of tasks  $\mathcal{T}_i \sim p(\mathcal{T})$

4:     **for all**  $\mathcal{T}_i$  **do**

5:         Evaluate  $\nabla_{\theta} \mathcal{L}_{\mathcal{T}_i}(f_{\theta})$  with respect to  $K$  examples

6:         Compute adapted parameters with gradient descent:  $\theta'_i = \theta - \alpha \nabla_{\theta} \mathcal{L}_{\mathcal{T}_i}(f_{\theta})$

7:     **end for**

8:     Update  $\theta \leftarrow \theta - \beta \nabla_{\theta} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta'_i})$

9: **end while**

---

# Meta-aprendizado em nível de instância

## Model Applicability Induction (MAI)

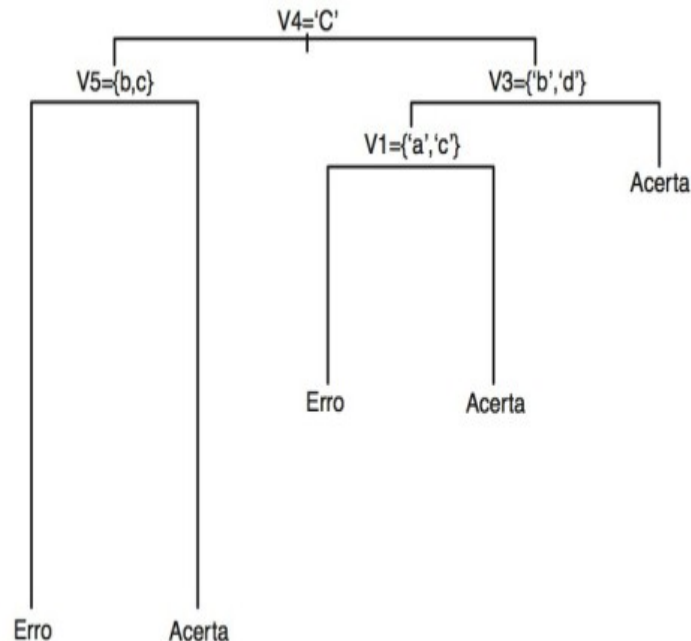
- Utiliza meta-aprendizado para determinar se utiliza ou não cada um dos modelos induzidos

V1	V2	V3	V4	V5	Classe	V1	V2	V3	V4	V5	Erro
t	a	c	t	a	membro	t	a	c	t	a	+
t	g	c	t	a	membro	t	g	c	t	a	-
g	t	a	c	t	não membro	g	t	a	c	t	+
a	a	t	t	g	membro	a	a	t	t	g	+
t	c	g	a	t	não membro	t	c	g	a	t	-
a	g	g	g	g	membro	a	g	g	g	g	+

# Meta-aprendizado em nível de instância

## Model Applicability Induction (MAI)

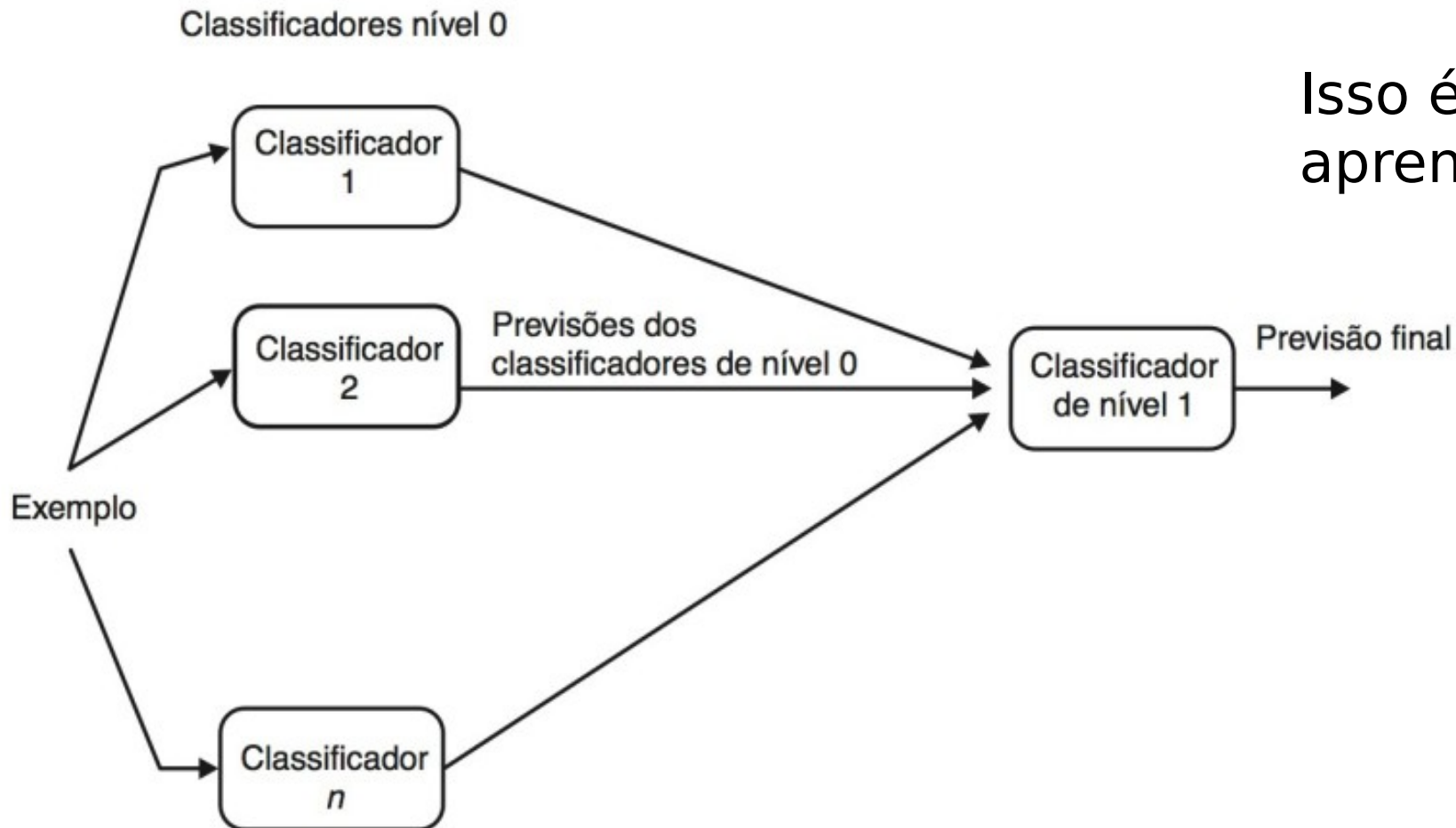
- Utiliza meta-aprendizado para determinar se utiliza ou não cada um dos modelos induzidos





# Meta-aprendizado em nível de instância

## *Stacking*



Isso é meta-aprendizado?

# Meta-aprendizado em nível de instância

## Stacking

Score do modelo 3  
para a classe 1

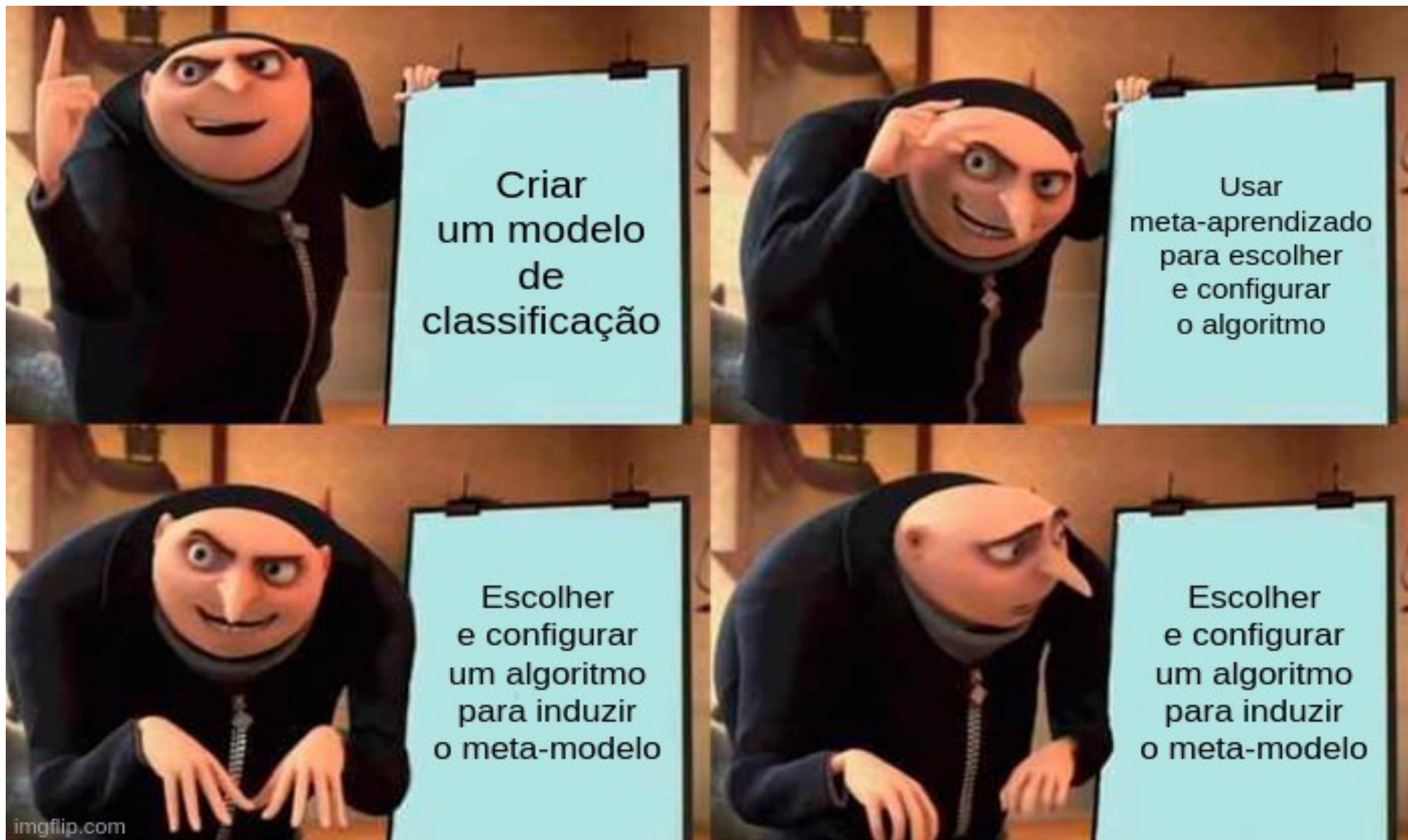
A diferença é a  
substituição da  
classe pela meta-  
classe!

V1	V2	V3	V4	V5	Classe	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	$P_{3,2}$	Classe
t	a	c	t	a	Membro	0,51	0,49	0,13	0,87	0,12	0,88	Membro
t	g	c	t	a	Membro	0,19	0,81	0,07	0,93	0,81	0,19	Membro
g	t	a	c	t	Não Membro	0,68	0,32	0,55	0,45	0,69	0,31	Não Membro
a	a	t	t	g	Membro	0,74	0,26	0,66	0,34	0,94	0,06	Membro
t	c	g	a	t	Não Membro	0,62	0,38	0,01	0,99	0,78	0,22	Não Membro
a	g	g	g	g	Membro	0,65	0,35	0,90	0,10	0,55	0,45	Membro

Conjunto de dados original

Conjunto de dados de  $Nível_1$

Por hoje... vocês já sabem



# Referências e bibliografia

- K. A. Smith-Miles, “Cross-disciplinary perspectives on meta-learning for algorithm selection,” ACM Comput. Surv., vol. 41, no. 1, pp. 1–25, Dec. 2008. para J. R. Rice, “The algorithm selection problem,” Adv. Comput., vol. 15, pp. 65–118, Jan. 1976.
- I. Khan, X. Zhang, M. Rehman and R. Ali, "A Literature Survey and Empirical Study of Meta-Learning for Classifier Selection," in IEEE Access, vol. 8, pp. 10262-10281, 2020, doi: 10.1109/ACCESS.2020.2964726.
- Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina Katti Faceli, Ana Carolina Lorena, João Gama, Tiago Agostinho de Almeida e André C. P. L. F de Carvalho. Inteligência artificial: uma abordagem de aprendizado de máquina. Segunda edição: LTC, 2021. 304 p. ISBN 9788521637349.