

Bootcamp: Analista de Machine Learning

Trabalho Prático

Módulo 3	Seleção de Modelos de Aprendizado de Máquina
-----------------	---

Objetivos

O Trabalho Final em laboratório virtual possui como objetivos principais:

1. Utilizar **dois** algoritmos de classificação para resolver um problema.
2. Fazer a análise deste problema com uso de métricas de qualidade, comparando as soluções.
3. Tornar mais robusta a solução do problema com uso de validação cruzada e busca de hiperparâmetros.

Para este trabalho, a linguagem utilizada será o Python, e a ferramenta de desenvolvimento recomendada é o Google Colaboratory. O objetivo, em suma, é **gerar dois modelos** para resolver o problema e **compará-los usando as métricas de qualidade adequadas e técnicas para deixá-los robustos**.

Enunciado

Trabalho Final - O dataset IRIS

Amostras por classe: 50

Dimensionalidade: 4

Tipo das features: Números reais positivos

O conjunto de dados IRIS talvez seja o mais clássico dos conjuntos de testes de algoritmos de classificação. Ele divide amostras de plantas em três espécies diferentes e tem, como features, a altura e largura de duas partes da planta: a sépala e as pétalas.

Neste trabalho, você deve classificar esse conjunto de dados com pelo menos dois algoritmos. Você deverá analisá-los segundo as métricas de qualidade que julgar relevantes, e utilizar técnicas de busca de hiperparâmetros e validação cruzada para tornar seus resultados mais robustos.

Fonte: http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris.html

Observação: Gerar gráficos que expliquem e ilustrem o seu problema não é parte obrigatória da resolução do trabalho, mas é um treino bastante recomendado. Além disso, **será necessário consultar a documentação do scikit-learn** para fazer esse trabalho. Busque por “algoritmos de classificação” e pelos códigos mostrados durante as aulas interativas. Saber consultar documentações é uma habilidade importante de quem trabalha com Aprendizado de Máquina, e é isso que pretendemos desenvolver aqui!

Código Inicial

```
from sklearn import datasets
from sklearn.decomposition import PCA

iris = datasets.load_iris()
X = iris.data[:, :2] # pegamos as primeiras duas features aqui. Pegue as outras se quiser.
y = iris.target

y[0] # classe do primeiro elemento
X[1, :] # primeiro elemento a classificar
```

Rodando o trabalho

Você pode rodar o trabalho **da forma que quiser**, mas se não tiver preferência por como fazê-lo, recomendamos o uso do **Google Colaboratory**. São poucos passos para utilizá-lo:



Welcome To Colaboratory

File Edit View Insert Runtime Tools Help



Table of contents



Getting started

Data science



Machine learning

More Resources

Machine Learning Examples

Section

+ Code

+ Text

Copy to Drive



What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in y

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can mak
just get started below!

Getting started

The document you are reading is not a static web page, but an interactive env
code.

A página inicial do **Google Colaboratory** se parece com essa. Clique em *File* e clique em *New notebook*.



Welcome To Colaboratory

File Edit View Insert Runtime Tools Help



Table of contents



Getting started

Data science



Machine learning

More Resources

Machine Learning Examples

Section

New notebook

Open notebook

⌘/Ctrl+O

Upload notebook

Rename notebook

Move to trash

Save a copy in Drive

Save a copy as a GitHub Gist

Save a copy in GitHub

Save

⌘/Ctrl+S

Save and pin revision

⌘/Ctrl+M S

Revision history

Download .ipynb

Download .py

+ Code

+ Text

Copy to Drive



What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in you

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make
just get started below!

Getting started

The document you are reading is not a static web page, but an interactive enviro

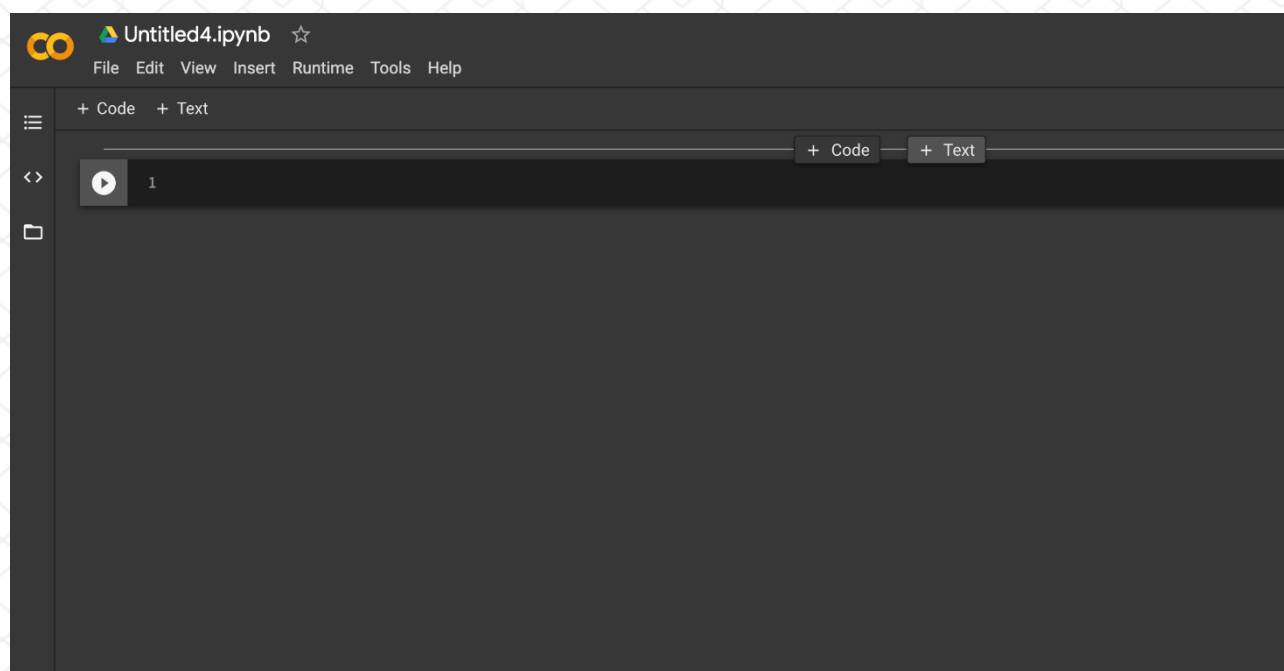
Para prosseguir, você precisará logar com uma conta Google. Faça isso.

Google sign-in required

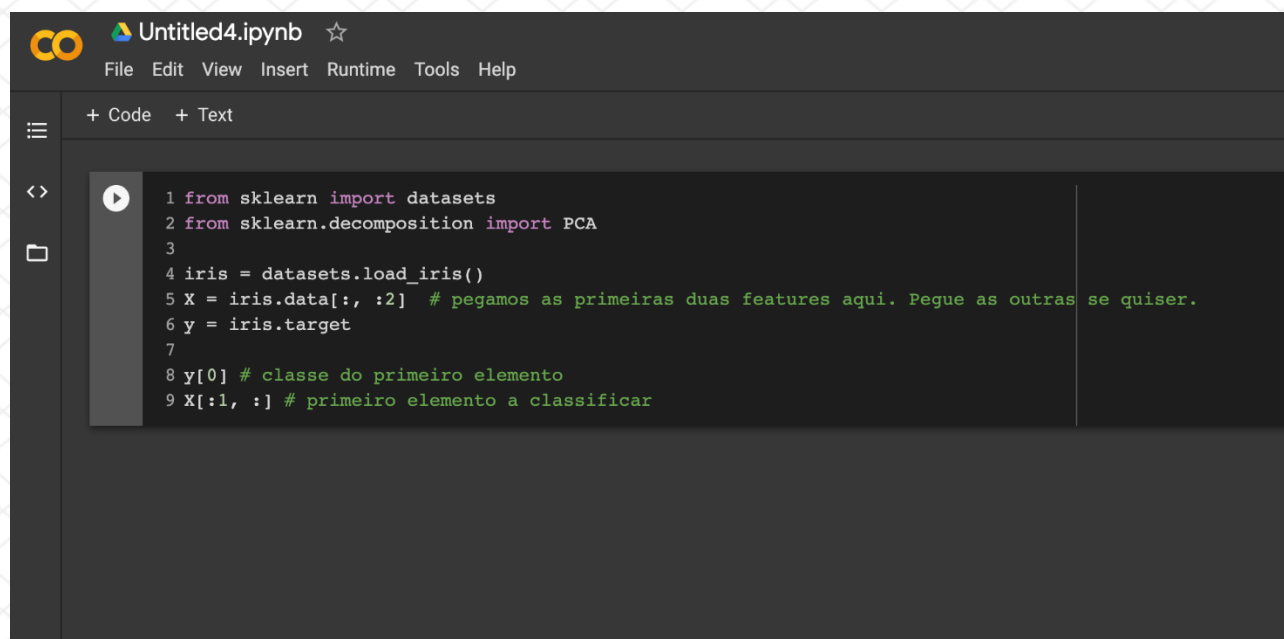
You must be logged in with a Google Account to continue.

OK

A seguir, você verá uma tela parecida com essa. É só clicar na caixa no centro da tela e escrever seu código.



A seguir, basta apertar o botão de *play*, ao lado esquerdo da caixa, e seu código rodará. **Atenção! Ele só imprimirá alguma coisa na tela se você escrever no programa algo que o faça (como o comando `print`)!**



The screenshot shows a Jupyter Notebook titled 'Untitled4.ipynb'. The interface includes a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. Below the menu bar, there are tabs for '+ Code' and '+ Text'. The code cell is active, displaying the following Python code:

```
1 from sklearn import datasets
2 from sklearn.decomposition import PCA
3
4 iris = datasets.load_iris()
5 X = iris.data[:, :2] # pegamos as primeiras duas features aqui. Pegue as outras se quiser.
6 y = iris.target
7
8 y[0] # classe do primeiro elemento
9 X[1, :] # primeiro elemento a classificar
```

Respostas Finais

Os alunos deverão desenvolver a prática e, depois, responder às seguintes questões objetivas: