MLflow

Mlflow é um software que auxilia o desenvolvimento de ML

Dois conceitos importantes no mlflow são os experimentos e as execuções

Os experimentos podem representar um trabalho ou um desenvolvimento de um modelo especifico, um experimento reune varias execuções

Ja as execuções são, as execuções do seu modelo, as execuções tem atreladas a elas diversas informações

- **Versão do código** : Git commit usado para executar a execução se ela foi executada a partir de um projeto do MLflow.
- · Início e fim : Hora de início e hora de término da execução
- **Origem**: nome do arquivo executado para iniciar a execução, ou o nome do projeto e o ponto de entrada da execução, se a execução foi executada a partir de um projeto do MLflow.
- **Parâmetros** : parâmetros de entrada de valor-chave de sua escolha. Chaves e valores são cadeias de caracteres.
- **Métricas**: métricas de valor-chave em que o valor é numérico. Cada métrica pode ser atualizada ao longo da execução (por exemplo, para rastrear como a função de perda do seu modelo está convergindo), e o MLflow registrará e permitirá visualizar o histórico completo da métrica.
- Artefatos : arquivos de saída em qualquer formato. Por exemplo, você pode gravar imagens (por exemplo, PNGs), modelos (por exemplo, um modelo SciKit-Learn em conserva) ou até arquivos de dados.

Para utilizar o ml flow

- 1. Impota-se o a biblioteca mlflow
- 2. Cria-se um experimento ou se seleciona um ja existente
- 3. Inicia uma execução
- 4. Na execução se salva as metricas, parametros, artefatos e etc
- 5. Se encerra a execução
- 6. Utiliza da interface para consultar os dados salvos

Para criar um experimento se usa o comando

 $mlflow.create_experiment(name)$

Para usar um experiemento que ja existe se usa

mlflow.set_experiment(experiment_name)

Para associar os registros a um experimento e iniciar/terminar uma execução

 ${\tt mlflow.start_run}(run_id=None,\ experiment_id=None,\ run_name=None)$ ${\tt mlflow.end_run}()$

Para salvar metricas, parametros e etc

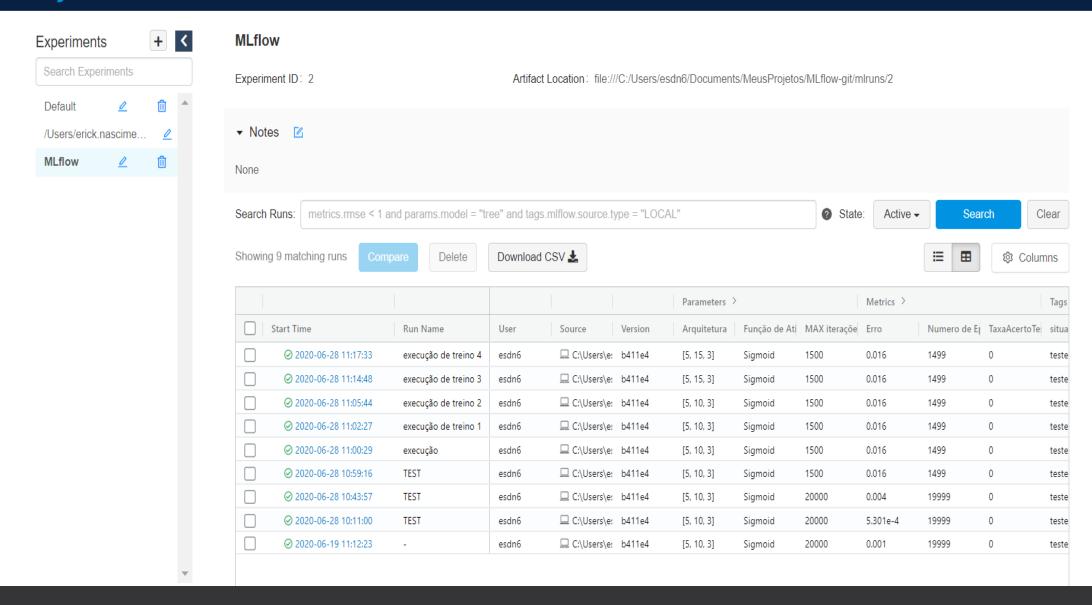
mlflow.log_x

Quando se cria um experimento pela primeira vez, um diretorio mlruns é criado, neste esta salvo todos os dados dos experimentos, e com ele um experimento Default esse exeperimento recebe todas as execuções em que não seja expecificado outro experimento

Para acessar a interface grafica do mlflow basta executar o comando mlflow ui no diretorio contendo o mlruns

Depois no navegador acessar http://localhost:5000/

ml*fl*ow





MLflow > Run ade37a49f39d4fa1af34c07f0b8f0fb9 -

Date: 2020-06-19 11:12:23

User: esdn6

Source:

 $\begin{tabular}{ll} \square C:\Users\esdn6\Documents\MeusProjetos\MLflow-git\MLP\ com\ mlflow \\ \end{tabular} \begin{tabular}{ll} Git\ Commit: b411e4177bd2dd837993f93690326a4fa2415410 \\ \end{tabular}$

(1).py

Duration: 8.5min

Status: FINISHED

▼ Notes

None

▼ Parameters

Name	Value
Arquitetura	[5, 10, 3]
Função de Ativação	Sigmoid
MAX iterações	20000
Momentum	0.5
Redução Minima	1e-11
Taxa de Aprendizagem	0.1

▼ Metrics

MAX iterações	20000
Momentum	0.5
Redução Minima	1e-11
Taxa de Aprendizagem	0.1

▼ Metrics

Name	Value
Erro 🗠	0.001
Numero de Epocas 🗠	19999
TaxaAcertoTeste 🗠	0
Tempo de Treinamento 🗠	507.5

▼ Tags

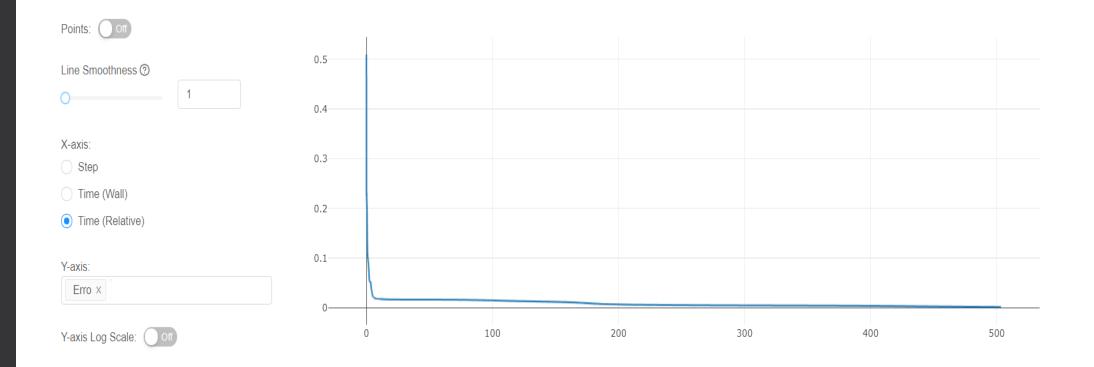
Name	Value	Actions
situação	teste	<u>_</u>

Add Tag

Name Value Add

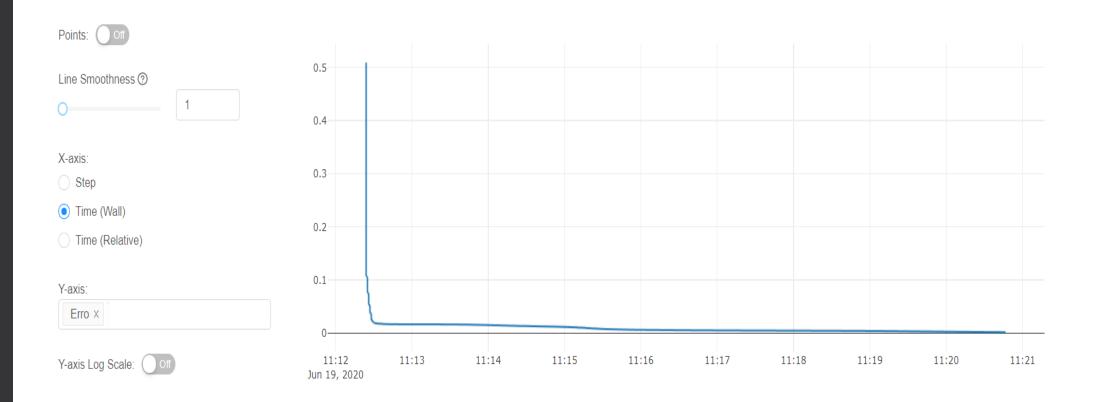


MLflow > Run ade37a49f39d4fa1af34c07f0b8f0fb9 > Erro





MLflow > Run ade37a49f39d4fa1af34c07f0b8f0fb9 > Erro



Pesos da camada de saida.csv

3 Saidas de teste.csv

Pesos da primeira camada de escondida.csv

B Pesos iniciais da primeira camada de escondid

Pesos iniciais da camada de saida.csv

Entradas de teste.csv

Full Path: file:///C:/Users/esdn6/Documents/MeusProjetos/MLflow-git/mlruns/2/ade37a49f39d4fa1af34c07f0b8f0fb9/artifa...

↓

neuronio1, neuronio2, neuronio3

- 0.2199485793977548,0.8371409616743847,0.6239415925369487
- 0.1265354585302243,0.12368431142379943,0.10082633217706327
- 0.28183199692560335,0.47243015785444553,0.6293661976311423
- 0.9583043199396009,0.2551017666363935,0.4671129509744363
- 0.02359359868384403,0.9256032785564557,0.8890945326980835
- 0.7675963483941793,0.8404180007377707,0.9243356487901502
- 0.8465314415268843,0.18857513035441154,0.6786441299791349
- 0.9662131536546431,0.07224649237485092,0.35134515716824966
- 0.7472226721327601,0.4868305406698987,0.22671402111531558
- 0.40822922592047806,0.21388341574702052,0.768617940488647

Alem disso o mlflow possui integração com algumas bibliotecas de machin e learning o que permite salvar modelos ja configurados e treinados como artefatos, esses modelos podem ser carregados para serem reutilizados.

Por exemplo em pode se salvar um modelo sklearn com o comando

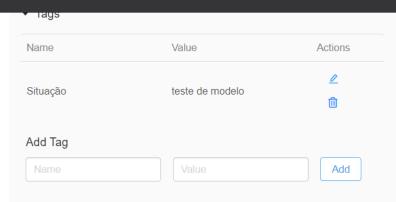
mlflow.sklearn.log_model(model, "model") /

mlflow.sklearn.save_model(model, "model")

E com o comando

mlflow.sklearn.load_model(model_uri)

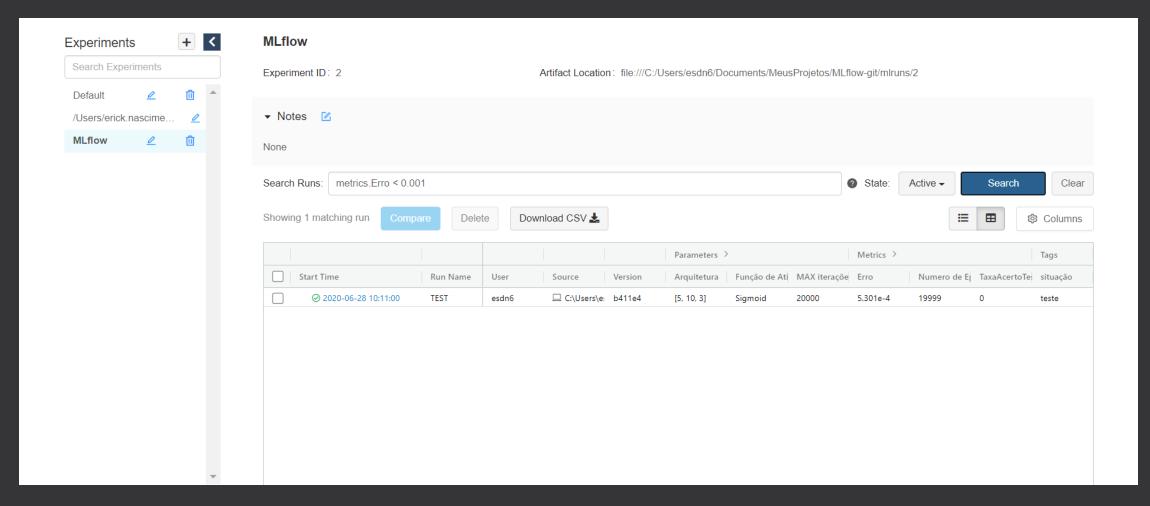
Para usar um modelo ja salvo



Artifacts



Durante o desenvolvimento de machine learning é comum ter diversos modelos e execuções diferentes com resultados diversos então a interface do mlflow permite a pesquisa de execuções baseados em atributos das execuções



A pesquisa tambem é possivel com o comando mlflow.search_runs(experiment_ids='x')

Esse comando retorna um dataframe contendo todas as runs do experimento especificado

-	df_runs = mlflow.search_runs(exp df_runs	eriment_ids=						
2]:	run_id	experiment_id	status	artifact_uri	start_time	end_time	metrics.Numero de Epocas	metrics.TaxaAcerto
•	f46e308468414ab58296bb3a53b6f0e6	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 19:27:52.473000+00:00	2020-06-28 19:28:13.498000+00:00	9999.0	
1	1 0bcb8f773a824e279e9c224d80c53d14	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 19:26:26.766000+00:00	2020-06-28 19:26:30.825000+00:00	999999.0	
2	2 9c2421b934f642d6b3f221beef2acf18	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:39:57.481000+00:00	2020-06-28 19:26:24.450000+00:00	999999.0	
3	3 677470f2852147e5a5456442b8367edd	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:35:04.822000+00:00	2020-06-28 18:37:40.204000+00:00	99999.0	
4	4 4932851583804d618513b09c16f86123	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:34:03.688000+00:00	2020-06-28 18:34:11.628000+00:00	4999.0	
	97e60df3fc4a4484822e78db3c336189	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:32:59.091000+00:00	2020-06-28 18:32:59.289000+00:00	49.0	
•	6 0e088229b34b4597b6eecfbd05026554	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:31:15.147000+00:00	2020-06-28 18:31:38.478000+00:00	14999.0	
7	7 724b50e1bc39462ca7a6bbd04f15060b	1	FINISHED	file:///home/erick/MLF/mlflow%20pesquisa/mlrun	2020-06-28 18:30:02.034000+00:00	2020-06-28 18:30:05.328000+00:00	1499.0	
8	rows × 21 columns							

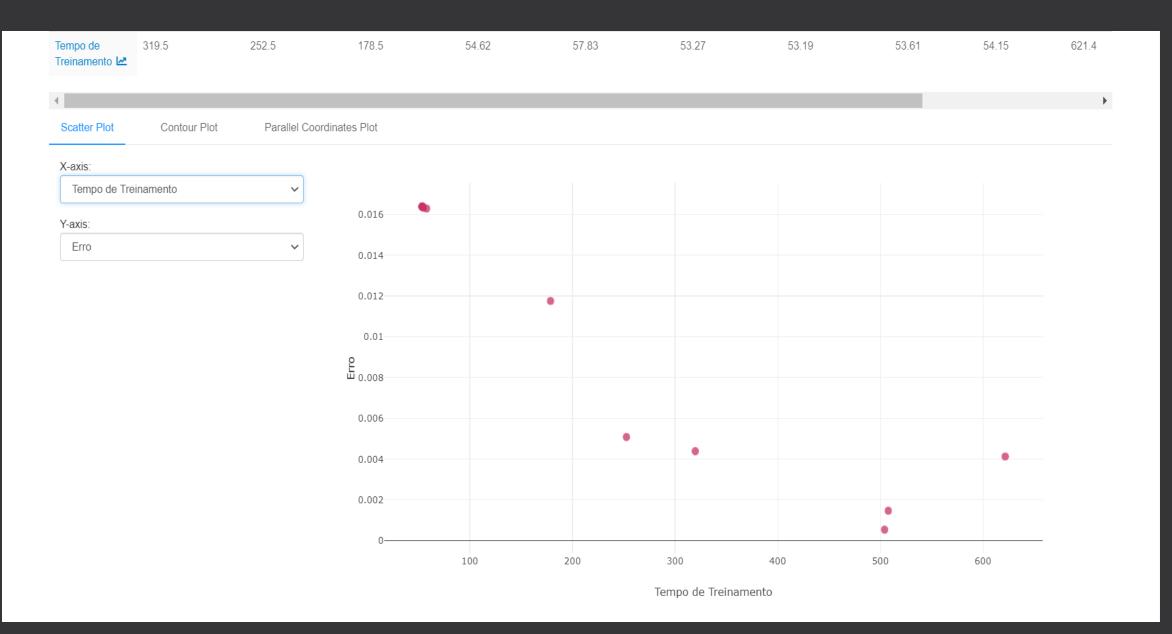
Tambem é possivel selecionar e comparar execuções, por diversas metricas.

Por exemplo podendo escolher modelos não aena baseado em um parametro podendo traçar um grafico de Erro X Tempo de Previsão escolhendo de forma equilibrada entre desenpenho e taxa de acerto



MLflow > Comparing 12 Runs

Run ID:	f0c8052f1df64f4f9	ffef486ecf054f299	85996d0a7ac2437	168bad108c08470	92cc28aa635e4ca	f8a965289d6e4ec	dc3cea8a3eb54b3	60e1bf0dcf30	c93fda605d68	d05a90e
Run Name:	execução de treino 4	execução de treino 3	execução de treino 2	execução de treino 1	execução	TEST	TEST			
Start Time:	2020-06-28 18:06:42	2020-06-28 18:00:46	2020-06-28 17:56:26	2020-06-28 11:17:33	2020-06-28 11:14:48	2020-06-28 11:05:44	2020-06-28 11:02:27	2020-06-28 11:00:29	2020-06-28 10:59:16	2020-06- 10:43:57
Parameters										
Arquitetura	[5, 15, 3]	[5, 15, 3]	[5, 15, 3]	[5, 15, 3]	[5, 15, 3]	[5, 10, 3]	[5, 10, 3]	[5, 10, 3]	[5, 10, 3]	[5, 10, 3]
Função de Ativação	Sigmoid	Sigmoid	Sigmoid							
MAX iterações	10000	7500	5000	1500	1500	1500	1500	1500	1500	20000
Momentum	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
Redução Minima	1e-16	1e-16	1e-16	1e-11	1e-11	1e-11	1e-11	1e-11	1e-11	1e-11
Taxa de Aprendizagem	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Metrics										
Erro 🗠	0.004	0.005	0.012	0.016	0.016	0.016	0.016	0.016	0.016	0.004
Numero de Epocas ∠	9999	7499	4999	1499	1499	1499	1499	1499	1499	19999



Os modelos salvos tem o que documentação chama de flavor, que se refere a origem do codigo por exemplo do sklearn ou do tensorflow

Um dos flavors é o python function (pyfunc) que permite salvar um "modelo" python generico

Tambem é possivel com comando **mlflow models serve**, servir modelos e solicitar previsões, essas podem ser solicitadas pelo comando **curl**

```
C:\Users\esdn6\Documents\Projeto\model sklearn\mlruns\0\494f234817ff404b979c816e19ebd103\artifacts>mlflow models serve -m model
2020/07/02 22:06:56 INFO mlflow.models.cli: Selected backend for flavor 'python_function'
2020/07/02 22:07:31 INFO mlflow.pyfunc.backend: === Running command 'conda activate mlflow-ddd5f319eedafda373e2c515826c087b2afd202d & waitres
s-serve --host=127.0.0.1 --port=5000 --ident=mlflow mlflow.pyfunc.scoring_server.wsgi:app'
c:\users\esdn6\anaconda3\envs\mlflow-ddd5f319eedafda373e2c515826c087b2afd202d\lib\site-packages\waitress\adjustments.py:441: DeprecationWarni
ng: In future versions of Waitress clear_untrusted_proxy_headers will be set to True by default. You may opt-out by setting this value to Fal
se, or opt-in explicitly by setting this to True.
warnings.warn(
Serving on http://erickn:5000
```

```
esdn6@erickn MINGW64 ~/Documents/Projeto
$ curl -d '{"columns":[0],"index":[0,1],"data":[[11],[-10]]}' -H 'Content-Type: application/json' localhost:5000/invocations
% Total % Received % Xferd Average Speed Time Time Current
Dload Upload Total Spent Left Speed
100 90 100 41 100 49 62 75 --:--:-- --:-- 138[22.000000000000004, -20.0000000000000]
```