

---

# **cognite-model-hosting-notebook Documentation**

*Release 0.1.0*

**Cognite**

**Apr 24, 2019**



---

# Contents

---

<b>1</b>	<b>Jupyter Notebook Utilities</b>	<b>3</b>
1.1	Deploy only . . . . .	3
1.2	Train and deploy . . . . .	6
1.3	Local Artifacts . . . . .	7
<b>2</b>	<b>Exceptions</b>	<b>9</b>



Utilites for deploying a model to Cognite Model Hosting environment from a Jupyter Notebook.

To install this package run the following command

```
pip install cognite-model-hosting-notebook
```



## 1.1 Deploy only

```
cognite.model_hosting.notebook.notebook.deploy_model_version (name: str,
                                                                model_id: int, runtime_version: str,
                                                                artifacts_directory: Optional[str] =
                                                                None, description: Optional[str] =
                                                                None, metadata: Optional[Dict[str,
                                                                str]] = None, notebook_path: Optional[str] =
                                                                None, cognite_client: Optional[cognite.client.cognite_client.Cognite
                                                                = None) → int
```

Deploy the model version in the current notebook to the model hosting environment.

### Parameters

- **name** (*str*) – The name of the model version.
- **model\_id** (*int*) – Id of the model to deploy the version to.
- **runtime\_version** (*str*) – The model hosting runtime version to deploy the model to.
- **artifacts\_directory** (*str, optional*) – Path of the directory containing any artifacts you want to include with your deployment.
- **description** (*str, optional*) – Description of this model version.
- **metadata** (*Dict[str, str], optional*) – Any metadata to include about this model version.

- **notebook\_path** (*str*, *optional*) – The path to the notebook. If omitted, the notebook you're in will be used.
- **cognite\_client** (*CogniteClient*, *optional*) – The CogniteClient instance to use for uploading the model. If omitted, a new instance will be created using the API key in the COGNITE\_API\_KEY environment variable.

**Returns** The model version id

**Return type** int





## 1.2 Train and deploy

```
cognite.model_hosting.notebook.notebook.train_and_deploy_model_version(name:  
    str,  
    model_id:  
    int,  
    run-  
    time_version:  
    str,  
    de-  
    scrip-  
    tion:  
    Op-  
    tional[str]  
    =  
    None,  
    meta-  
    data:  
    Op-  
    tional[Dict[str,  
    str]]  
    =  
    None,  
    args:  
    Op-  
    tional[Dict[str,  
    Any]]  
    =  
    None,  
    scale_tier:  
    Op-  
    tional[str]  
    =  
    None,  
    ma-  
    chine_type:  
    Op-  
    tional[str]  
    =  
    None,  
    note-  
    book_path:  
    Op-  
    tional[str]  
    =  
    None,  
    cog-  
    nite_client:  
    Op-  
    tional[cognite.client.cognite_  
    =  
    None)  
    →  
    int
```

Train and deploy the model version in the current notebook in the model hosting environment.

**Parameters**

- **name** (*str*) – The name of the model version.
- **model\_id** (*int*) – Id of the model to deploy the version to.
- **runtime\_version** (*str*) – The model hosting runtime version to deploy the model to.
- **description** (*str, optional*) – Description of this model version.
- **metadata** (*Dict[str, str], optional*) – Any metadata to include about this model version.
- **args** (*Dict[str, Any], optional*) – Arguments to pass to the train function defined on your model.
- **scale\_tier** (*str, optional*) – Scale tier to train on. Must be “CUSTOM” or “BASIC”.
- **machine\_type** (*str, optional*) – Machine type to use. Only applicable if `scale_tier` is “CUSTOM”.
- **notebook\_path** (*str, optional*) – The path to the notebook. If omitted, the notebook you’re in will be used.
- **cognite\_client** (*CogniteClient, optional*) – The `CogniteClient` instance to use for uploading the model. If omitted, a new instance will be created using the API key in the `COGNITE_API_KEY` environment variable.

**Returns** The model version id

**Return type** `int`

## 1.3 Local Artifacts

`cognite.model_hosting.notebook.notebook.local_artifacts` (*model\_version\_name: str, root\_dir: Optional[str] = None*) → Callable

Local artifacts storage.

Returns a function which works like the builtin `open()`, but pointing to a directory specific to the provided model version name. The directory will be `artifacts/<model_version_name>`. By default the artifacts directory will reside in the current working directory.

**Parameters**

- **model\_version\_name** (*str*) – The name of the model version which the artifacts belong to.
- **root\_dir** (*str, optional*) – The root directory where the *artifacts* directory reside. Defaults to the current working directory.

**Returns**

A function/context manager which works like the builtin *open*. Let’s your read and write to the local artifacts directory.

**Return type** Callable

## Examples

Using `local_artifacts()`:

```
open_artifact = local_artifacts('my-model')
with open_artifact('my_file.txt', 'w') as f:
    f.write('A cool file\n') # will be stored in artifacts/my-model/my_file.txt
```

## CHAPTER 2

---

### Exceptions

---

**exception** `cognite.model_hosting.notebook.notebook.UnsupportedNotebookVersion`  
Raised if the current version of Jupyter Notebook is not supported by this integration.

**exception** `cognite.model_hosting.notebook._model_file.InvalidCodeFormat`  
Raised if there is a mistake in how you have defined your model within the notebook.

**exception** `cognite.model_hosting.notebook._setup_file.InvalidRequirements`  
Raised if your cell defining requirements is not correctly formatted.



## D

`deploy_model_version()` (in module `cognite.model_hosting.notebook.notebook`), 3

## I

`InvalidCodeFormat`, 9

`InvalidRequirements`, 9

## L

`local_artifacts()` (in module `cognite.model_hosting.notebook.notebook`), 7

## T

`train_and_deploy_model_version()` (in module `cognite.model_hosting.notebook.notebook`), 6

## U

`UnsupportedNotebookVersion`, 9