[[model API]](https://keras.io/models/about-keras-models/)

* **model.layers** is a flattened list of the layers comprising the model.
* **model.inputs** is the list of input tensors of the model.
* **model.outputs** is the list of output tensors of the model.
* **model.summary()** prints a summary representation of your model. Shortcut for utils.print\_summary
* **model.get\_config()** returns a dictionary containing the configuration of the model. The model can be reinstantiated from its config via:

*config = model.get\_config()*

*model = Model.from\_config(config)*

*# or, for Sequential:*

*model = Sequential.from\_config(config)*

* **model.get\_weights()** returns a list of all weight tensors in the model, as Numpy arrays.
* **model.set\_weights(weights)** sets the values of the weights of the model, from a list of Numpy arrays. The arrays in the list should have the same shape as those returned by get\_weights().
* **model.to\_json()** returns a representation of the model as a JSON string. Note that the representation does not include the weights, only the architecture. You can reinstantiate the same model (with reinitialized weights) from the JSON string via:

*from keras.models import model\_from\_json*

*json\_string = model.to\_json()*

*model = model\_from\_json(json\_string)*

* **model.to\_yaml()** returns a representation of the model as a YAML string. Note that the representation does not include the weights, only the architecture. You can reinstantiate the same model (with reinitialized weights) from the YAML string via:

*from keras.models import model\_from\_yaml*

*yaml\_string = model.to\_yaml()*

*model = model\_from\_yaml(yaml\_string)*

* **model.save\_weights(filepath)** saves the weights of the model as a HDF5 file.
* **model.load\_weights(filepath, by\_name=False)** loads the weights of the model from a HDF5 file (created by save\_weights). By default, the architecture is expected to be unchanged. To load weights into a different architecture (with some layers in common), use by\_name=True to load only those layers with the same name.