- JSON format
    - .json for json
    - .ubj for binary json

- Booster.save_model(‘model.json’)

- Since XGB is a gradient boosting library, there are two main parts
    - Model consisting of trees
    - Hyperparameters and configurations used for building the model
    - When the above function is called, XGB saves the trees, some parameters, and the objective function
    - The objective controls global bias (base score)

- Serialisation - The process of converting an object’s state into a format that can be stored or transmitted
- In cases where something more than the model needs to be saved (like if you want to continue computing in a new framework), the serialisation output is necessary
- This is called a memory snapshot

Backward Compatibility

- Models can be saved for long term storage using save_model
- Models use stable representation so are accessible in later versions of XGB
- Memory snapshots are not stable and my not be accessible in later versions

- Custom functions (user provided, language dependent features) are not stable and may not work on different versions of python or XGB

- Pickled file - stores a serialized representation of a Python object
- Pickled modules may be valuable if loaded back into the specific version of Python and XGB