# ML Deployment – Sports News Headlines Classifier

* Build Package:
  + Project folder path: C:\Users\Aiden\Documents\Data\_Science\_Stuff\sf\_Data\_Science\_Stuff\Projects\06\_ML\_Deployment\01\_Build\_Package
  + I have taken a copy of the code from the actual project’s folder path:
  + C:\Users\Aiden\Documents\Data\_Science\_Stuff\sf\_Data\_Science\_Stuff\Projects\01\_SportsNewsClassifier\final\_code
  + I have changed some of the code:
    - Code to process data and build model is saved in logistic\_regression\_model folder.
    - instead of scraping data, we load CSVs – **this is to ensure the best practice of reproducibility in our ML pipeline**
    - I have created requirements.txt and test\_requirements.txt for installing dependencies via Tox (we are not using conda!!)
    - I have created validation.py and predict.py that makes a prediction on a batch of data
    - I have used pytest instead of unittest. Tests are saved in the folder tests. The conftest.py loads the testing dataset which is used on the pytests.
  + Training data = Sports\_News\_09\_09\_2021.csv
  + Testing data = Sports\_News\_03\_06\_2021.csv
  + As mentioned above, I am using tox to run the pipeline:
  + tox.ini file contains commands to run the pipeline
  + most of the code was copied from udemy course section 5 tox.ini
  + I also copied over the mypy.ini file as these remove warnings when running commands in tox.ini.
  + pytest was producing deprecation warnings so I added another file called pytest.ini to remove these warnings when running tox
  + At this point the package is ready to be built
  + I copied over the pyproject.toml file from Udemy and at the bottom I changed the line\_length to 135
  + I also copied over setup.py and made changes:
    - Package meta-data
    - Instead of pathlib, I have used os to locate directories
  + I created a file called VERSION that specifies the version of the model.
  + Lastly, I built the python package for the model (tid-logistic-regression-model)
    - I added init.py file to the folder logistic\_regression\_model. Without this file, the package will not work in the next section
* **FastAPI**
  + FastAPI loads after running tox -e run
    - Ensure correct version of model is in requirements.txt
  + localhost:8001 to view app
  + /docs to execute test prediction
  + Can also use requests module in Python to make a prediction:
* **Deploy to Heroku**
  + Create Heroku app from cli: peaceful-fjord-46734
  + At this point we will need a git repository (create development branch in git and push all code there!)
  + git subtree push --prefix 02\_Deploy\_REST\_API main
  + this should deploy the app – go back to Heroku and open app. Can check out /docs and execute predicton.