Automating code deployment in Windows

One interesting challenge to find was how to automate my modeling tasks so I could instead spend my time on more interesting, cognitive pursuits.

Data scientists like me spend a lot of time discussing the business problem and developing a modeling solution. Once the data is available and has been processed for modeling, the next step involves trying and testing a few algorithms and evaluating the findings with business intuition.

At some point when the model outcomes have been shared with stakeholders and approved, the model moves into maintenance or deployment or production stage (the exact label name varies by organization or teams). In this stage, the model needs to be run on a pre-determined agreed upon frequency (hourly/daily/weekly/monthly/quarterly). This would be best served by “automating” the task of model predictions. This is also called developing the modeling pipeline that produces results on a recurring and reproducible basis.

This repo walks through the steps needed to automate this task using just a windows machine and using Windows Subsystem for Linux WSL (<https://docs.microsoft.com/en-us/windows/wsl/about>). It would be useful to install WSL for this exercise (<https://docs.microsoft.com/en-us/windows/wsl/install>). This requires admin privileges or atleast someone with privileges to install on your behalf.

In this example, the housing data from Kaggle (<https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data>) is used to develop a basic illustrative model. This task would be automated using Powershell or Task Scheduler in Windows.

Files in the folder:

data\_description.txt – Explains the source features

train.csv – contains attributes for 1460 houses along with their sales prices

test.csv – contains attributes for 1460 houses whose sales prices need to be estimated

housing.py – the python file that needs to be executed automatically

scheduling.pdf – File with Powershell commands and Windows Task Scheduler screenshots to schedule our python code

In real life data science applications, we would need to update the training data in this folder daily before our predetermined task execution time. The model predictions will get stored in our predefined output file (which I prefer to time stamp as shown in the code).

This repo is not intended towards making the best modeling predictions or inference. It focuses instead on how to develop an automated process to be deployed in windows local. In a future repo, we can show the automation process in AWS as well as MS Azure. One advantage of using the cloud services is that the code

While this is an example for a data science model deployment using python, you can use these steps to automate pretty much all modeling tasks using any language as long as Windows has the necessary software to compile.