



Lesson 11

CI/CD

Now, automate it!

Learn About

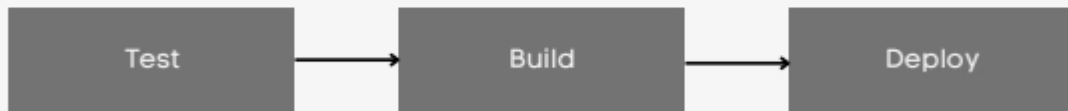


- Basics of Git and GitHub
- Introduction to CI/CD
- CI/CD in Machine Learning
- Jenkins



Introduction to CI/CD

- Abbreviation for **Continuous Integration/Continuous Delivery.**
- A CI/CD pipeline is generally composed of 3 steps



Git & GitHub

Criteria	Git	GitHub
Product	Git is a command-line tool	GitHub is a service with GUI
Installation	Git is installed locally	GitHub is hosted on the web
Maintainers	Git is maintained by linux	GitHub is maintained by Microsoft
Purpose	Git is focused on version control and code sharing	GitHub is focused on centralized source code hosting



A basic workflow with Git

1. Tell Git to work with your repo
 - `git init`
2. After making some changes, tell Git where
 - `git add .`
 - **OR** `git add file_1.txt file2/`
3. Add a message to make your changes understandable
 - `git commit -m "Added a unit test for model prediction"`
4. Create your repository on GitHub named `house_price_prediction`
5. Add a remote GitHub repo to push your code to
 - `git remote add origin https://github.com/quan-dang/house_price_prediction.git`
(please change quan-dang to your GitHub username)



A basic workflow with Git

6. Create and switch to the main branch

- `git checkout -b main`

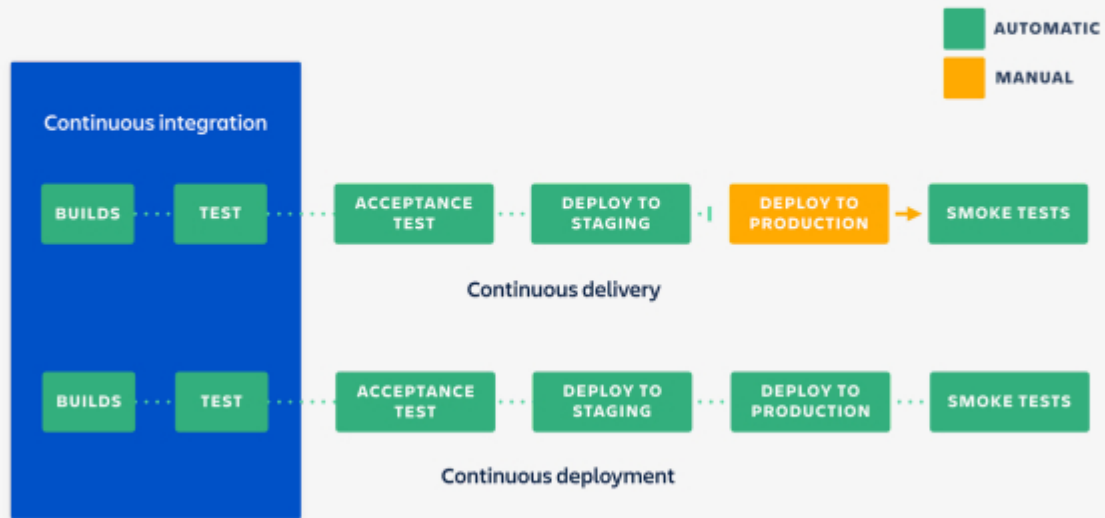
7. Push to the current branch

- `git push origin main`

8. Make some other changes, then repeat the steps 2., 3., and 7.



Delivery vs Deployment

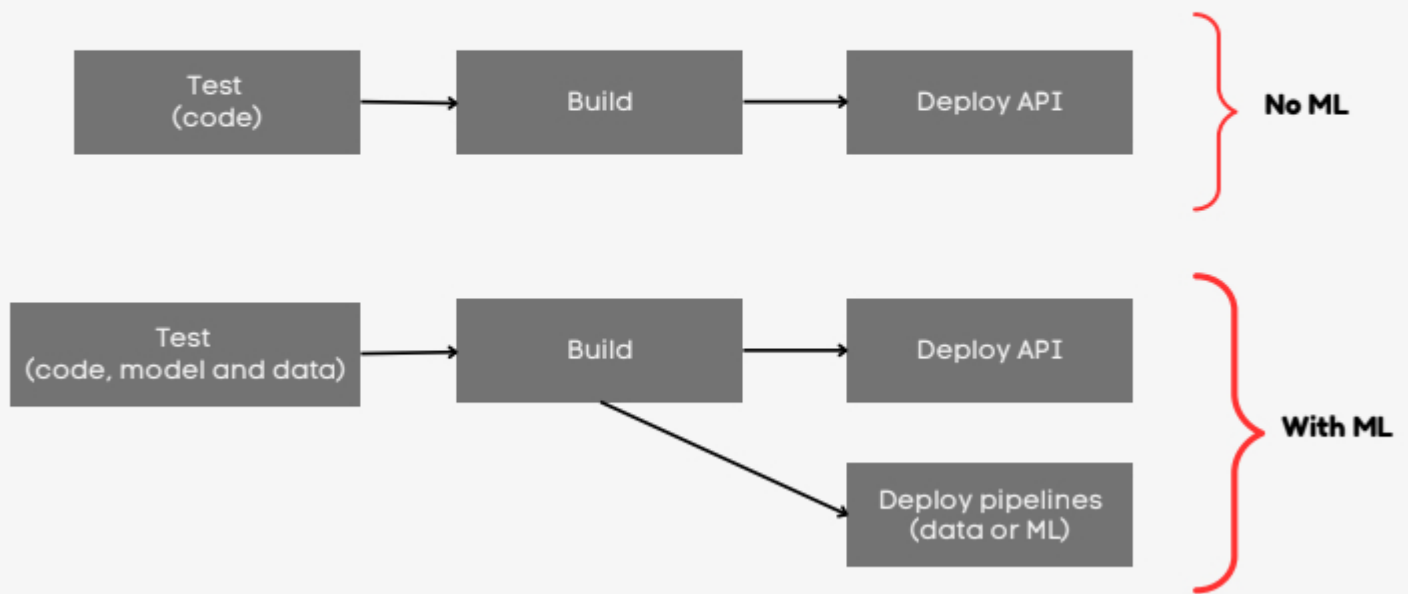


Popular CI/CD tools

	Gitlab	Bitbucket	GitHub	Jenkins
CI or CD	Both	Both	Both	Both
<u>Pricing</u>	Mid	The cheapest	Most expensive	Free
Free private repo	Yes	Yes	Yes	No, it's not for storing your code
<u>Open Source</u>	Yes, but can not deploy on Windows. and <u>limited features</u>	No	No	Yes, apply to most of GNU/Linux distros and windows
Project Analytics & Burndown chart	Yes	Yes	No	No



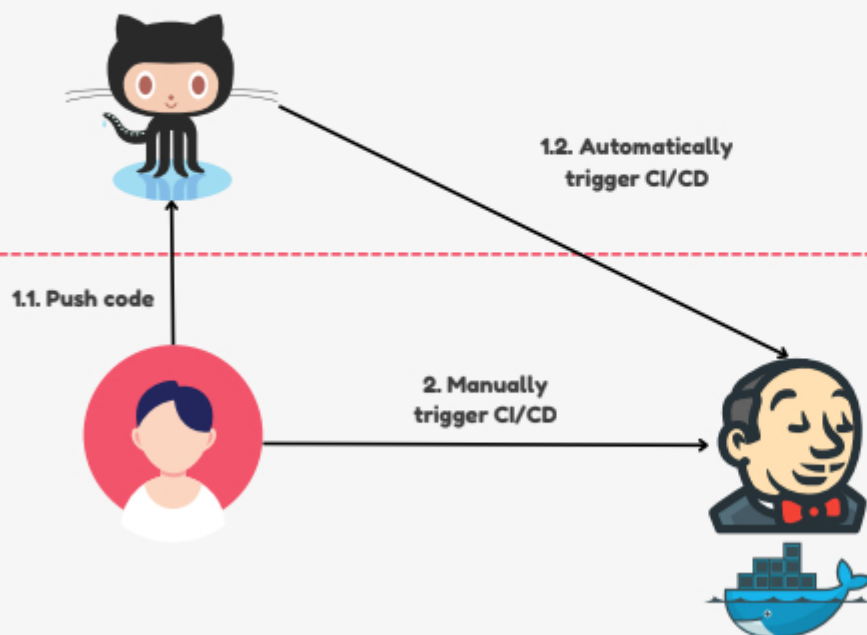
CI/CD in Machine Learning



Jenkins

Remote

Local



Jenkinsfile

jenkins_tutorial > Jenkinsfile-helloworld

```
1 pipeline {
2 |   // Any available agent ip:port or an docker image
3   agent any
4
5   stages {
6     stage('Test') {
7       // Only run when it has some changes in a file
8       // when { changeset "main.py" }
9
10      // Only run when it has some changes in a branch
11      // when { branch 'master' }
12
13      steps {
14        echo 'Testing something..'
15      }
16    }
17    stage('Build') {
18      steps {
19        echo 'Building something..'
20      }
21    }
22    stage('Deploy') {
23      steps {
24        echo 'Deploying something..'
25      }
26    }
27  }
28 }
29 }
```

Three stages in the pipeline

Write any commands you want

Thank You!



Created by
Quan Dang



