

# HW1

## Task1 GIT Command

### Github setup

For MAC user, just use terminal;

For Windows user, please install GitHubSetup.exe

### Basic command

```
cd aml_hw1
```

```
git init
```

```
git remote add origin
```

```
https://github.com/AppliedMachineLearning/homework-i-joannahu.git
```

\*before re-type: 

```
git remote rm origin
```

```
git clone https://github.com/AppliedMachineLearning/homework-i-joannahu.git
```

\*if it's a private repository: 

```
git clone https://joannahu:ss219177@github.com/AppliedMachineLearning/homework-i-joannahu.git
```

```
https://<account\_name>:<password>@github.com/AppliedMachineLearning/homework-i-joannahu.git
```

```
cd homework-i-joannahu/
```

```
echo "UNI:qh2174" >> README.md
```

```
git add README.md
```

```
git commit -m "add UNI"
```

```
git push -u origin master
```

```
mkdir Task1
```

```
git add Task1
```

```
git commit -m 'add folder'
```

```
git push -u origin master
```

\*undo a git push, you need to 'force' push the old reference.

```
git push -f origin last_known_good_commit:branch_name
```

### Task 1

```
mkdir aml_hw1
```

```
cd aml_hw1
```

```
git init
```

```
Administrator@CHINA-20141012Y /e/Documents/GitHub/aml_hw1 (master)
```

```
$ for num in 1 2 3 4 5
```

```
> do
```

```

> touch $num
> git add $num
> git commit -m "commit file $num"
> done
[master (root-commit) e47861f] commit file 1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1
[master 52eaac6] commit file 2
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 2
[master e7a13ac] commit file 3
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 3
[master e2aed61] commit file 4
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 4
[master 28e2d5d] commit file 5
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 5

```

```

$ git log --graph --oneline --decorate --all
* 28e2d5d (HEAD -> master) commit file 5
* e2aed61 commit file 4
* e7a13ac commit file 3
* 52eaac6 commit file 2
* e47861f commit file 1

```

git branch feature HEAD~4 #create new branch at 1 (4 commits before current HEAD)

\*delete: git branch -d the\_local\_branch

git checkout feature

```
$ for num in 6 7 8
```

```
> do
```

```
> touch $num
```

```
> git add $num
```

```
> git commit -m "commit file $num"
```

```
> done
```

```

[feature 8905b3b] commit file 6
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 6

```

```

[feature ce4f8ca] commit file 7
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 7

```

```

[feature 99e0bf7] commit file 8
1 file changed, 0 insertions(+), 0 deletions(-)

```

```

create mode 100644 8
git checkout master    #move to master branch
git rebase HEAD~2 --onto feature    #detach 4 and rebase to feature branch
$ git reflog
728e32b HEAD@{0}: rebase finished: returning to refs/heads/master
728e32b HEAD@{1}: rebase: commit file 5
8c63f0a HEAD@{2}: rebase: commit file 4
99e0bf7 HEAD@{3}: rebase: checkout feature
28e2d5d HEAD@{4}: checkout: moving from feature to master
99e0bf7 HEAD@{5}: commit: commit file 8
ce4f8ca HEAD@{6}: commit: commit file 7
8905b3b HEAD@{7}: commit: commit file 6
e47861f HEAD@{8}: checkout: moving from master to feature
28e2d5d HEAD@{9}: commit: commit file 5
e2aed61 HEAD@{10}: commit: commit file 4
e7a13ac HEAD@{11}: commit: commit file 3
52eaac6 HEAD@{12}: commit: commit file 2
e47861f HEAD@{13}: commit (initial): commit file 1

git checkout HEAD@{11}
touch 9
git add 9
git commit -m "commit file 9"
git branch debug
*if create debug before commit, you should checkout

git checkout master 7    #checkout
git add 7
git commit --amend --no-edit    #use commit message of 9

```

## Task2 Travis

How to write test function in python:

<http://doc.pytest.org/en/latest/getting-started.html>

在 Github 添加 Travis

put .yml file and requirements.txt in the root directory. (Conclude the configuration of python 2.7, 3.4, 3.5)

Setting→Integration & Service→Add service , choose “Travis CI”→Add  
click profile link -> 左侧选择 Applied machine learning

滑动开关

查看测试运行状态及结果:

[https://travis-ci.com/AppliedMachineLearning/<your\\_repo\\_name>](https://travis-ci.com/AppliedMachineLearning/<your_repo_name>)

## Task3 Sphinx

Sphinx is used to generate HTML documentation.

Tutorial: <http://chase-seibert.github.io/blog/2015/08/14/getting-started-with-sphinx.html>

It employs latex to write maths equations. For example,

.. math::

$$\text{mean} = \frac{\sum x_i}{N}$$
$$\text{std} = \sqrt{\frac{\sum (x_i - \text{mean})^2}{N}}$$

Input numpy array, when axis\_arg=0, calculate mean/std by columns

when axis\_arg=1, calculate mean/std by rows

when axis\_arg=None, calculate mean/std for all values

```
\begin{array}{ccc}
```

```
x_1 & x_2 & \dots \\
```

```
x_3 & x_4 & \dots \\
```

```
\vdots & \vdots & \ddots
```

```
\end{array}
```

The Task 3 and Task 4 result shows at

<https://appliedmachinelearning.github.io/homework-i-joannahu/>

## Task4 Visualization

Examples:

[https://appliedmachinelearning.github.io/homework-i-Jenseits14/html/task4/iris\\_api.html](https://appliedmachinelearning.github.io/homework-i-Jenseits14/html/task4/iris_api.html)

<https://appliedmachinelearning.github.io/homework-i-xinweili/>

