

Commands used for deploying the flask application on the ec2 instance

The following screenshots show the codes ran in ec2 instance for installing the docker and building a container and deploying the app in the container

Age Prediction

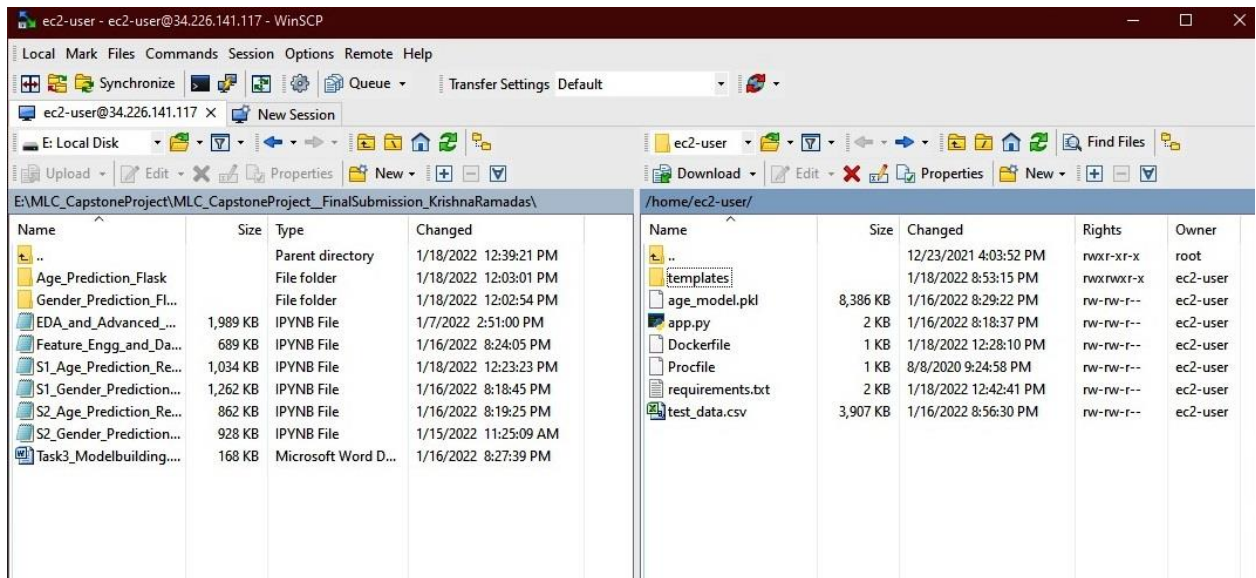
1. Installing docker

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-82-197 ~]$ sudo amazon-linux-extras install docker
Installing docker
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.7 kB 00:00
amzn2extra-docker | 3.0 kB 00:00
amzn2extra-kernel-5.10 | 3.0 kB 00:00
(1/7): amzn2-core/2/x86_64/group_gz | 2.5 kB 00:00
(2/7): amzn2-core/2/x86_64/updateinfo | 432 kB 00:00
(3/7): amzn2extra-kernel-5.10/2/x86_64/primary_db | 5.8 MB 00:00
(4/7): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 76 B 00:00
(5/7): amzn2extra-docker/2/x86_64/updateinfo | 4.7 kB 00:00
(6/7): amzn2extra-docker/2/x86_64/primary_db | 86 kB 00:00
(7/7): amzn2-core/2/x86_64/primary_db | 59 MB 00:00
Resolving Dependencies
Running transaction check
```

2. Starting docker service and giving permission to ec2-user

```
[ec2-user@ip-172-31-82-197 ~]$ sudo yum install docker
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package docker-20.10.7-5.amzn2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-82-197 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-82-197 ~]$ sudo usermod -a -G docker ec2-user
[ec2-user@ip-172-31-82-197 ~]$
```

3. Transferring files from local to ec2-user using WINSCP



4. Building docker image

First this code is run

```
sudo chmod 666 /var/run/docker.sock
```

```
[ec2-user@ip-172-31-82-197 ~]$ docker build -t ageprediction .
Sending build context to Docker daemon 2.906GB
Step 1/10 : FROM python:3.7-slim
---> d3c9ad326043
Step 2/10 : WORKDIR /app/
---> Using cache
---> 6cd0038bc3ae
Step 3/10 : COPY requirements.txt /app/
---> b11df4b6dce9
Step 4/10 : RUN pip install -r ./requirements.txt
---> Running in 885faf0bb78e
Collecting aiobotocore==2.1.0
  Downloading aiobotocore-2.1.0.tar.gz (54 kB)
Collecting aiohttp==3.8.1
```

5. Running app

```
ec2-user@ip-172-31-90-210:~
---> Running in be532b30110f
Removing intermediate container be532b30110f
---> 147320acb815
Step 9/10 : CMD ["app.py"]
---> Running in 423ee65ca584
Removing intermediate container 423ee65ca584
---> cc0f081a2f2a
Step 10/10 : EXPOSE 5000
---> Running in 4cb362ab9429
Removing intermediate container 4cb362ab9429
---> 5ef5afa2284f
Successfully built 5ef5afa2284f
Successfully tagged ageprediction:latest
[ec2-user@ip-172-31-90-210 ~]$ docker run -p 5000:5000 ageprediction
* Serving Flask app "app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 997-620-465
```

Gender Prediction

Up until step 4 is repeated for gender prediction app deployment then the following is done:

Gender prediction image build

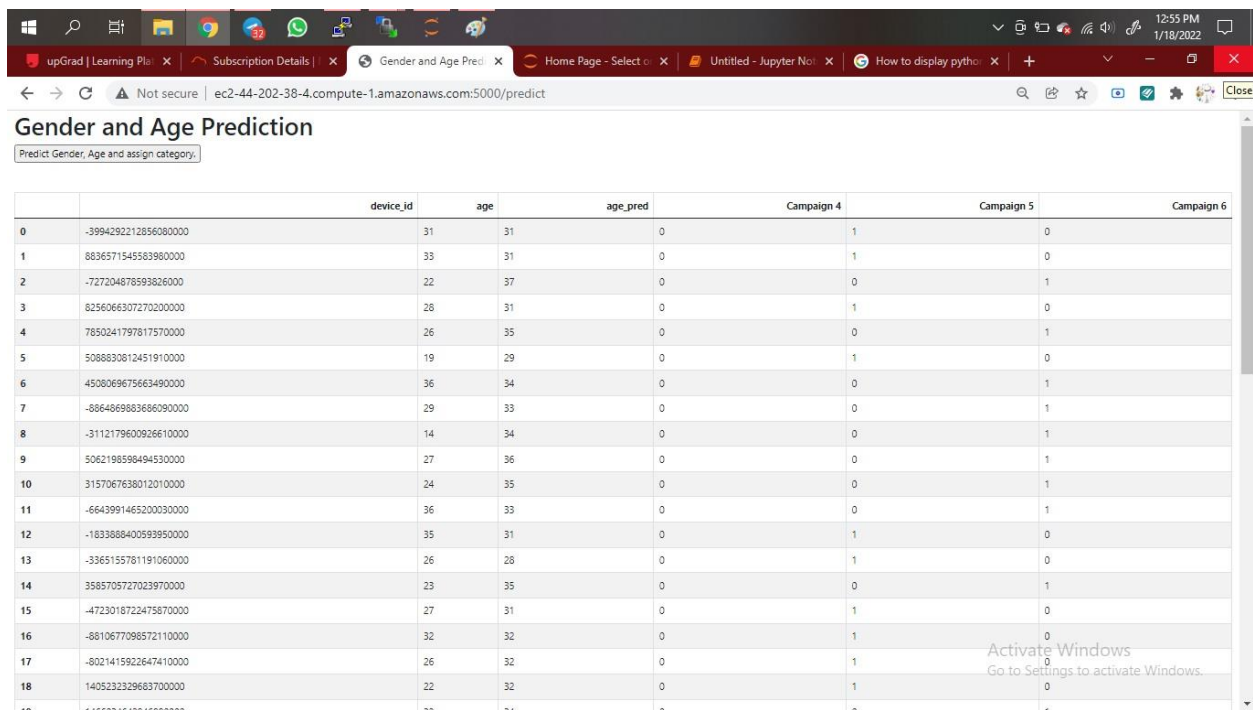
```
^C[ec2-user@ip-172-31-90-210 ~]$ docker build -t genderprediction .
Sending build context to Docker daemon  9.756MB
Step 1/10 : FROM python:3.7-slim
---> d3c9ad326043
Step 2/10 : WORKDIR /app/
---> Using cache
---> 8e17fb41b15a
Step 3/10 : COPY requirements.txt /app/
---> Using cache
---> abdl75b4a511
Step 4/10 : RUN pip install -r ./requirements.txt
---> Using cache
---> 680a72240bca
Step 5/10 : COPY app.py /app/
```


Deploying the app

```
Step 10/10 : EXPOSE 5000
---> Running in dflaaf108ad7
Removing intermediate container dflaaf108ad7
---> c55ad188b0de
Successfully built c55ad188b0de
Successfully tagged genderprediction:latest
[ec2-user@ip-172-31-90-210 ~]$ docker run -p 5000:5000 genderprediction
* Serving Flask app "app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 557-081-484
117.194.169.41 - - [19/Jan/2022 05:25:46] "POST /predict HTTP/1.1" 200 -
117.194.169.41 - - [19/Jan/2022 05:26:05] "POST /predict HTTP/1.1" 200 -
```

The screenshot of the final output for predicting the campaigns for the 50 customers

Age Prediction



	device_id	age	age_pred	Campaign 4	Campaign 5	Campaign 6
0	-3994292212856080000	31	31	0	1	0
1	863657154553980000	33	31	0	1	0
2	-727204878593826000	22	37	0	0	1
3	8256066307270200000	28	31	0	1	0
4	7850241797817570000	26	35	0	0	1
5	5088630612451910000	19	29	0	1	0
6	4508069675663490000	36	34	0	0	1
7	-686486983686090000	29	33	0	0	1
8	-3112179600926610000	14	34	0	0	1
9	5062198598494530000	27	36	0	0	1
10	3157067638012010000	24	35	0	0	1
11	-6643991465200300000	36	33	0	0	1
12	-1833886400593950000	35	31	0	1	0
13	-3365155781191060000	26	28	0	1	0
14	3585705727023970000	23	35	0	0	1
15	-4723018722475870000	27	31	0	1	0
16	-8810677098572110000	32	32	0	1	0
17	-8021415922647410000	26	32	0	1	0
18	1405232329663700000	22	32	0	1	0
19	-1462046400000000000	25	34	0	1	0

Gender Prediction

Gender and Age Prediction

Predict Gender, Age and assign category.

	device_id	gender	gender_prob	final_prediction	Campaign 1	Campaign 2	Campaign 3
0	-3994292212856080000	0	0.825882	1	1	0	0
1	8836571545583960000	0	0.797618	1	1	0	0
2	-727204878593826000	1	0.700102	1	0	0	0
3	8256066307270200000	0	0.709565	1	0	0	0
4	7850241797817570000	1	0.677176	1	0	0	0
5	5088830812451910000	1	0.713045	1	0	0	0
6	4508069675663490000	1	0.745365	1	1	0	0
7	-8864869883686090000	0	0.748732	1	1	0	0
8	-3112179600926610000	1	0.741170	1	1	0	0
9	5062198598494530000	1	0.658978	1	0	0	0
10	3157067638012010000	1	0.783116	1	1	0	0
11	-6643991465200030000	1	0.621858	0	0	0	0
12	-1833888400593950000	1	0.721961	1	0	0	0
13	-3365155781191060000	0	0.866136	1	1	0	0
14	3585705727023970000	1	0.629667	1	0	0	0
15	-4723018722475870000	1	0.815523	1	1	0	0
16	-8810677098572110000	1	0.737018	1	1	0	0
17	-8021415922647410000	1	0.695863	1	0	0	0
18	1405232329683700000	1	0.668466	1	0	0	0
19	-1162801819842000000	1	0.712262	1	1	0	0