- **Prepare the model (Using Toy Data)**

jupyter model Last Checkpoint: 4 days ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

[1]:
```
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.metrics import accuracy_score
from joblib import dump

# Load your dataset (replace with your actual file)
# Example: df = pd.read_csv("your_data.csv")
# For demonstration:
from sklearn.datasets import load_breast_cancer
data = load_breast_cancer()
X, y = pd.DataFrame(data.data, columns=data.feature_names), data.target

# Optional: reduce dimensionality (keep only top 15 features)
selector = SelectKBest(score_func=f_classif, k=15)
X_new = selector.fit_transform(X, y)

# Split data
X_train, X_test, y_train, y_test = train_test_split(X_new, y, test_size=0.2, random_state=42)

# Train a compact Random Forest model
model = RandomForestClassifier(n_estimators=25, max_depth=6, random_state=42)
model.fit(X_train, y_train)

# Test model performance (optional)
y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))

# Save model with compression
dump(model, "model.pkl", compress=4)  # compression level 0–9
print("Model saved as model.pkl (compressed).")
```

```
Accuracy: 0.956140350877193
Model saved as model.pkl (compressed).
```

- **Project Structure**

equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC: ~/Documents/Job/courses/Dat...

File Edit View Search Terminal Help

```
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ ls
app.py       model.joblib  render.yaml       templates
model.ipynb  model.pkl     requirements.txt
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ cd templates
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render/templates$ ls
index.html
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render/templates$
```

1. app.py

Procfile

```python
from flask import Flask, request, render_template
import joblib
import numpy as np

app = Flask(__name__)
model = joblib.load("model.joblib")

@app.route('/')
def home():
    return render_template("index.html")

@app.route('/predict', methods=['POST'])
def predict():
    features = [float(x) for x in request.form.values()]
    prediction = model.predict([features])[0]
    return render_template("index.html", prediction_text=f"Predicted class: {prediction}")

if __name__ == '__main__':
    app.run(debug=True)

```

2. templates/index.html

Open ~/Document

app.py | index.html

Name: ~/Documents/Job/courses/DataGlacier/flask_render/templates

MIME Type: plain text document (text/plain)
Encoding: Unicode (UTF-8)

```html
<!DOCTYPE html>
<html>
<head>
    <title>Prediction App</title>
</head>
<body>
    <h2>Enter input for prediction</h2>
    <form action="/predict" method="post">
        <input name="feature1" placeholder="Feature 1" required><br>
        <input name="feature2" placeholder="Feature 2" required><br>
        <input name="feature3" placeholder="Feature 3" required><br>
        <input name="feature4" placeholder="Feature 4" required><br>
        <input type="submit">
    </form>
    {% if prediction_text %}
    <h3>{{ prediction_text }}</h3>
    {% endif %}
</body>
</html>
```

5. requirements.txt

Open

requirements.txt
~/Documents/Job/courses/DataGlacier/flask_render

app.py | index.html | requirements.txt

```
1 flask
2 scikit-learn
3 joblib
4 gunicorn
```

6. render.yaml

Open

render.yaml
~/Documents/Job/courses/DataGlacier/flask_render

app.py | index.html | requirements.txt

```
services:
  - type: web
    name: flask-ml-app
    env: python
    buildCommand: "pip install -r requirements.txt"
    startCommand: "gunicorn app:app"
    plan: free
```

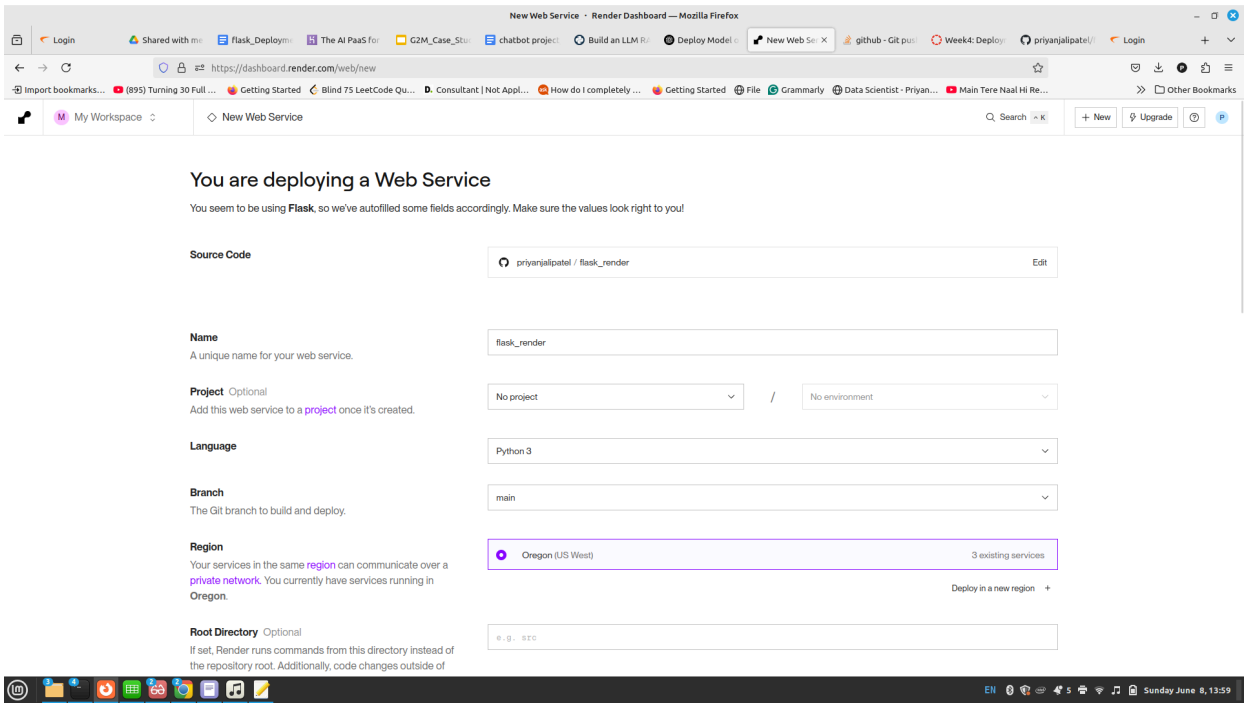
7. Push to GitHub

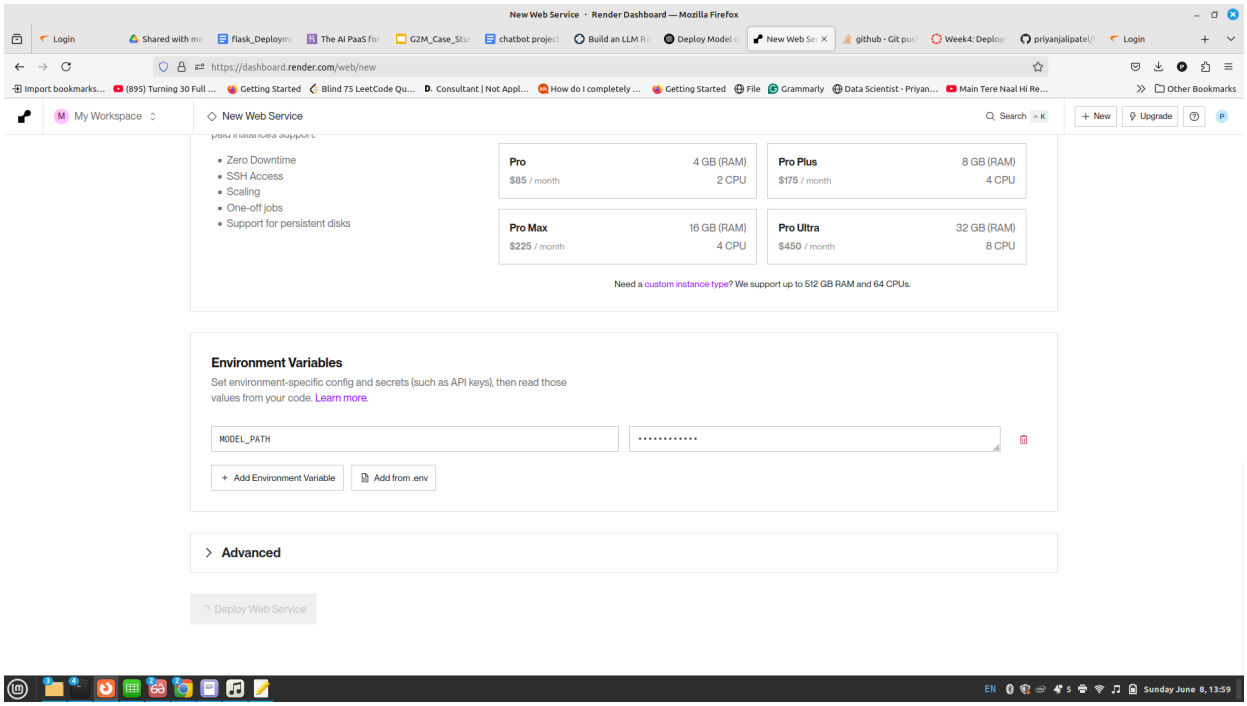
```
equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC: ~/Documents/Job/courses/DataGlacier/flask_render
File  Edit  View  Search  Terminal  Help
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git push
fatal: No configured push destination.
Either specify the URL from the command-line or configure a remote repository using

    git remote add <name> <url>

and then push using the remote name

    git push <name>

(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git commit -m "flask"
[master (root-commit) 1251372] flask
 7 files changed, 147 insertions(+)
 create mode 100644 app.py
 create mode 100644 model.ipynb
 create mode 100644 model.joblib
 create mode 100644 model.pkl
 create mode 100644 render.yaml
 create mode 100644 requirements.txt
 create mode 100644 templates/index.html
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git push
fatal: No configured push destination.
Either specify the URL from the command-line or configure a remote repository using

    git remote add <name> <url>

and then push using the remote name

    git push <name>

(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git push flask_render
fatal: 'flask_render' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git push priyanjalipatel/flask_render
fatal: 'priyanjalipatel/flask_render' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git remote add origin https://github.com/priyanjalipatel/fla
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git branch -M main
(base) equipo@equipo-ASUS-TUF-Dash-F15-FX517ZC-FX517ZC:~/Documents/Job/courses/DataGlacier/flask_render$ git push -u origin main
Username for 'https://github.com': priyanjali patel
Password for 'https://priyanjali patel@github.com':
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 16 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (10/10), 21.69 KiB | 10.84 MiB/s, done.
Total 10 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/priyanjalipatel/flask_render.git
 * [new branch]      main -> main
```

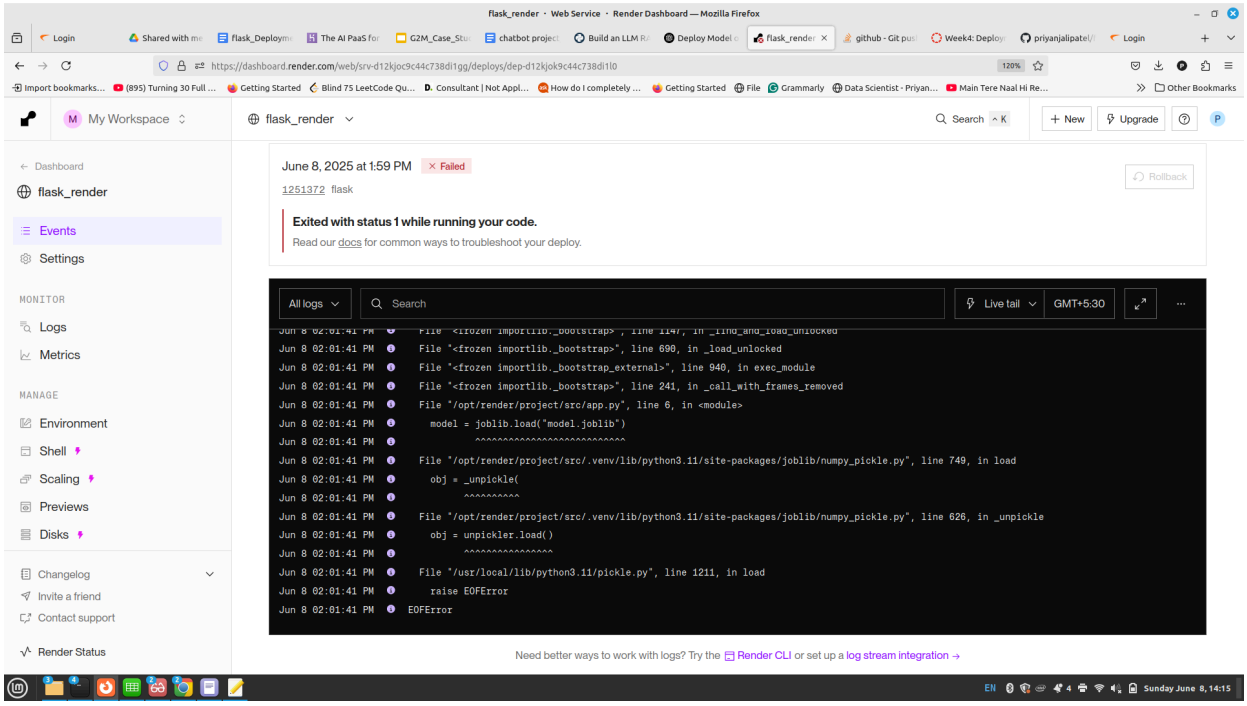
- **Deploy on Render**

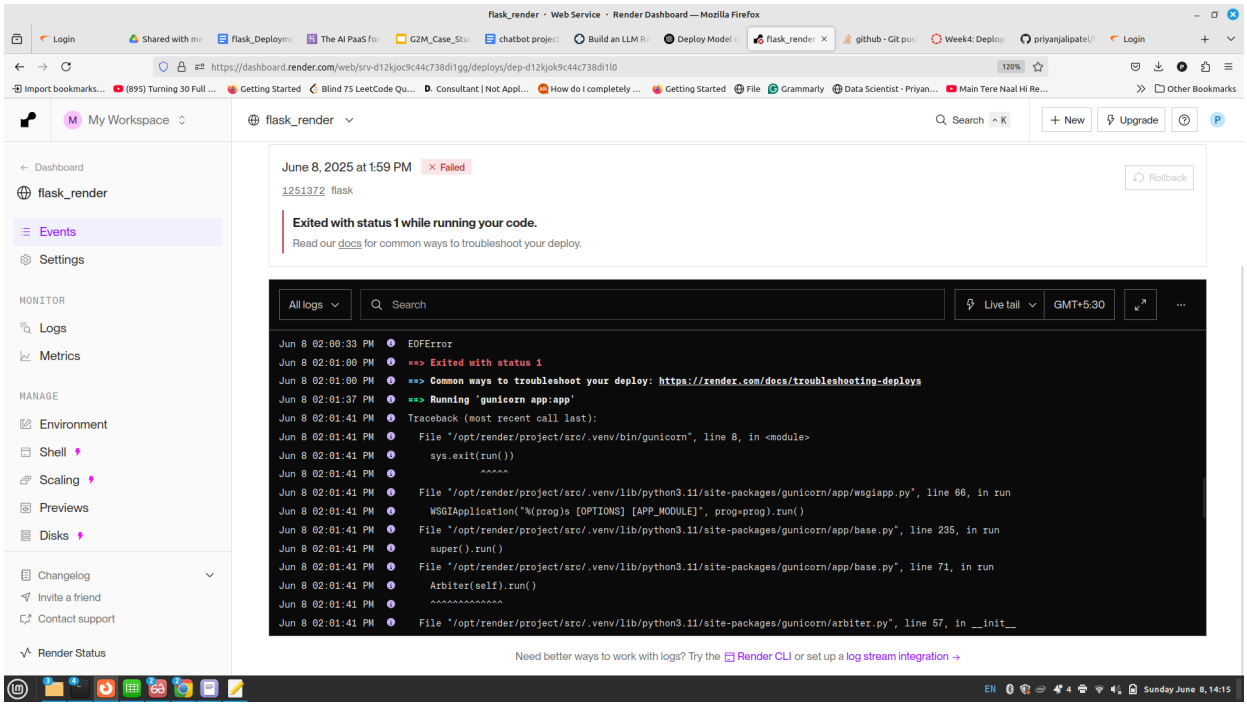
Use settings:

- Environment: Python
- Build Command: pip install -r requirements.txt
- Start Command: gunicorn app:app
- Free Plan




New Web Service · Render Dashboard — Mozilla Firefox
https://dashboard.render.com/web/new
My Workspace
New Web Service
Search
New
Upgrade
Zero Downtime
SSH Access
Scaling
One-off jobs
Support for persistent disks
Pro
$85 / month
4 GB (RAM)
2 CPU
Pro Plus
$175 / month
8 GB (RAM)
4 CPU
Pro Max
$225 / month
16 GB (RAM)
4 CPU
Pro Ultra
$450 / month
32 GB (RAM)
8 CPU
Need a custom instance type? We support up to 512 GB RAM and 64 CPUs.
Environment Variables
Set environment-specific config and secrets (such as API keys), then read those values from your code. Learn more.
MODEL_PATH
Add Environment Variable
Add from .env
Advanced
Deploy Web Service
Sunday June 8, 13:59


flask_render · Web Service · Render Dashboard — Mozilla Firefox
https://dashboard.render.com/web/srv-d12kjoc9c44c738di1gg/deploys/dep-d12kjok9c44c738di1l0
My Workspace
flask_render
Search
New
Upgrade
Dashboard
flask_render
Events
Settings
MONITOR
Logs
Metrics
MANAGE
Environment
Shell
Scaling
Previews
Disks
Changelog
Invite a friend
Contact support
Render Status
June 8, 2025 at 1:59 PM
Failed
1251372 flask
Rollback
Exited with status 1 while running your code.
Read our docs for common ways to troubleshoot your deploy.
All logs
Search
Live tail
GMT+5:30
Jun 8 02:01:41 PM File "<frozen importlib._bootstrap>", line 690, in _load_unlocked
Jun 8 02:01:41 PM File "<frozen importlib._bootstrap_external>", line 940, in exec_module
Jun 8 02:01:41 PM File "<frozen importlib._bootstrap>", line 241, in _call_with_frames_removed
Jun 8 02:01:41 PM File "/opt/render/project/src/app.py", line 6, in <module>
Jun 8 02:01:41 PM model = joblib.load("model.joblib")
Jun 8 02:01:41 PM ^^^^^^^^^^^^^^^^^^^^^^^^^^^
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/joblib/numpy_pickle.py", line 749, in load
Jun 8 02:01:41 PM obj = _unpickle(
Jun 8 02:01:41 PM ^^^^^^^^^^
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/joblib/numpy_pickle.py", line 626, in _unpickle
Jun 8 02:01:41 PM obj = unpickler.load()
Jun 8 02:01:41 PM ^^^^^^^^^^^^^^^^
Jun 8 02:01:41 PM File "/usr/local/lib/python3.11/pickle.py", line 1211, in load
Jun 8 02:01:41 PM raise EOFError
Jun 8 02:01:41 PM EOFError
Need better ways to work with logs? Try the Render CLI or set up a log stream integration →
Sunday June 8, 14:15


flask_render · Web Service · Render Dashboard — Mozilla Firefox
https://dashboard.render.com/web/srv-d12kjoc9c44c738di1gg/deploys/dep-d12kjok9c44c738di1l0
My Workspace
flask_render
Search
New
Upgrade
Dashboard
flask_render
Events
Settings
MONITOR
Logs
Metrics
MANAGE
Environment
Shell
Scaling
Previews
Disks
Changelog
Invite a friend
Contact support
Render Status
June 8, 2025 at 1:59 PM
Failed
1251372 flask
Rollback
Exited with status 1 while running your code.
Read our docs for common ways to troubleshoot your deploy.
All logs
Search
Live tail
GMT+5:30
Jun 8 02:00:33 PM EOFError
Jun 8 02:01:00 PM ==> Exited with status 1
Jun 8 02:01:00 PM ==> Common ways to troubleshoot your deploy: https://render.com/docs/troubleshooting-deploys
Jun 8 02:01:37 PM ==> Running 'gunicorn app:app'
Jun 8 02:01:41 PM Traceback (most recent call last):
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/bin/gunicorn", line 8, in <module>
Jun 8 02:01:41 PM sys.exit(run())
Jun 8 02:01:41 PM ^^^^^
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/gunicorn/app/wsgiapp.py", line 66, in run
Jun 8 02:01:41 PM WSGIApplication("%(prog)s [OPTIONS] [APP_MODULE]", prog=prog).run()
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/gunicorn/app/base.py", line 235, in run
Jun 8 02:01:41 PM super().run()
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/gunicorn/app/base.py", line 71, in run
Jun 8 02:01:41 PM Arbiter(self).run()
Jun 8 02:01:41 PM ^^^^^^^^^^^^
Jun 8 02:01:41 PM File "/opt/render/project/src/.venv/lib/python3.11/site-packages/gunicorn/arbiter.py", line 57, in __init__
Need better ways to work with logs? Try the Render CLI or set up a log stream integration →
Sunday June 8, 14:15