Model Deployment Heroku/Flask

Predict acceptance chance of university admission

LISUM07 – Andersson Andreé Romero Deza

24-03-2022

Agenda

Background

Model

Flask API

Heroku

Background

The data is from a finance company which provides various types of loans to its customer.

While auditing the data till previous year(2017), they found out that the salary is missing for some customers. It might be due to data entry issue or negligence by the loan manager.

Based on the loan history of the customers, help the company to predict the salary of the customers (whose salary details were missing) to maintain a good record for auditing purpose and future analysis.

Data source: https://www.kaggle.com/competitions/ml-l1-finalcs/overview/description

Create Flask API.

Heroku web.

Agenda

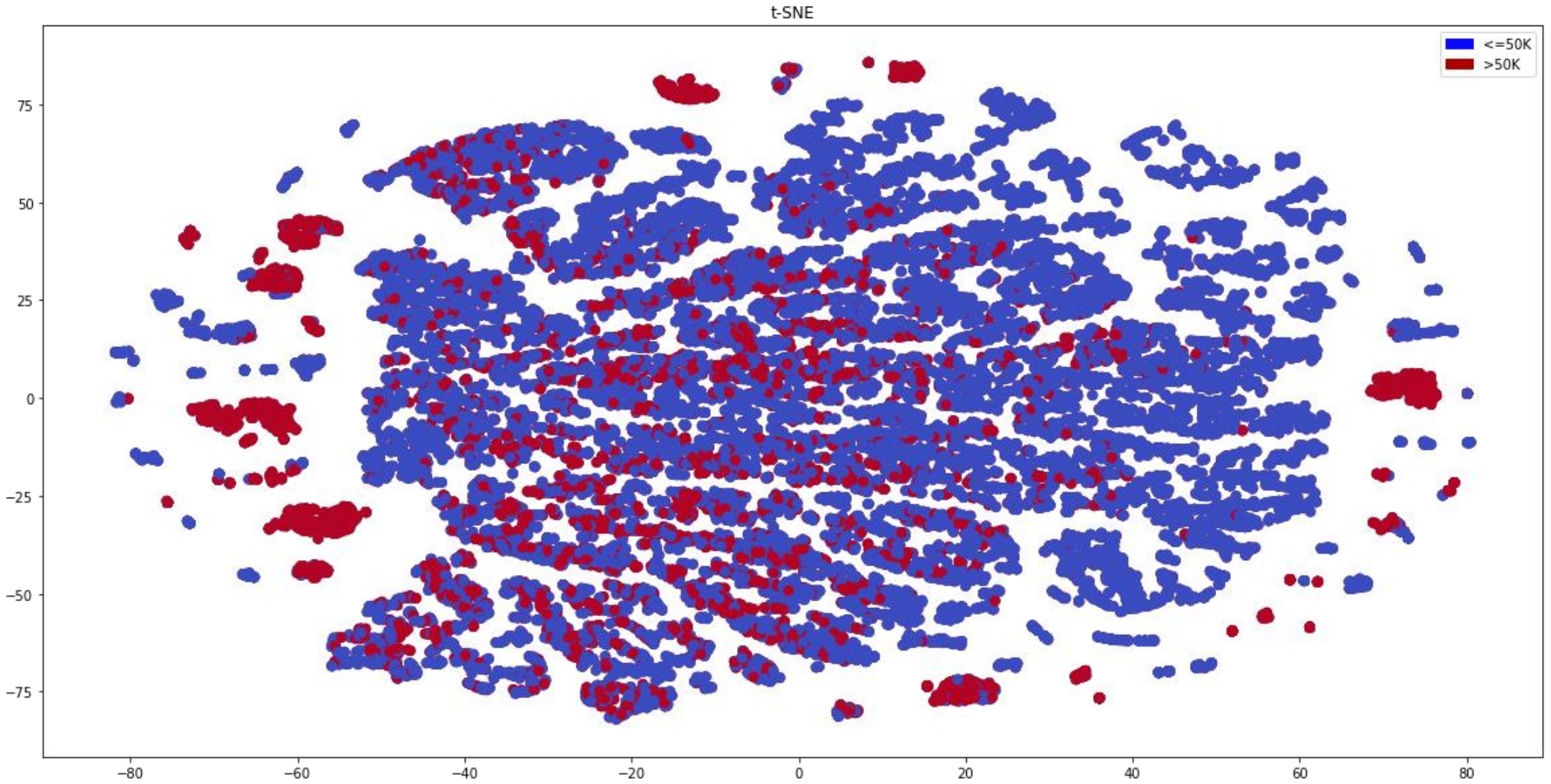
Background

Model

Flask API

Heroku

Model



Model

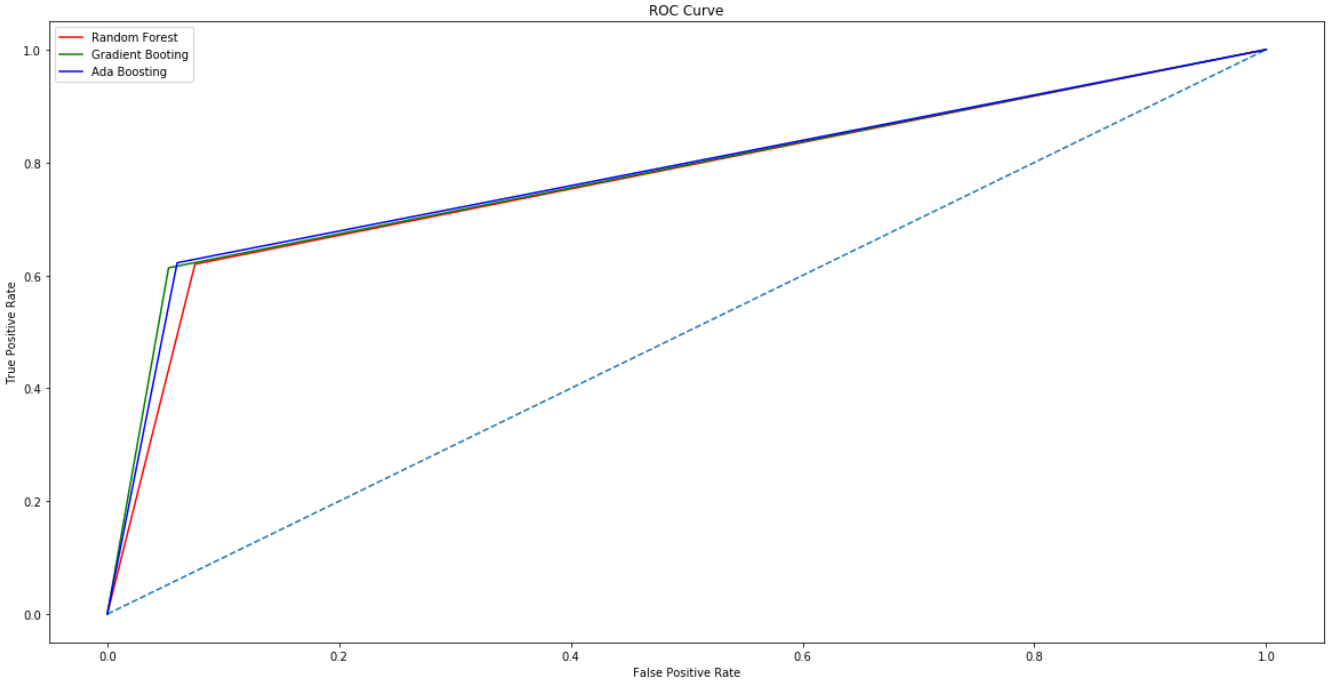
```
gbc = GradientBoostingClassifier(n_estimators=100, random_state=0)
gbc.fit(X_train, y_train)
ypred = gbc.predict(X_test)
print(confusion_matrix(y_test, ypred))
print(classification_report(y_test, ypred))
print("Accuracy Score:", accuracy_score(y_test, ypred))
print("Recall Score:", recall_score(y_test, ypred))
print("Precision Score:", precision_score(y_test, ypred))
print("ROC AUC Score: ", roc_auc_score(y_test, ypred))
gbc_fp, gbc_tp, gbc_threshold = roc_curve(y_test, ypred)
print("Threshold:", gbc_threshold)
```

```
[[7035  393]
 [ 905 1436]]
              precision    recall  f1-score   support

           0       0.89      0.95      0.92      7428
           1       0.79      0.61      0.69      2341

    accuracy                           0.87      9769
   macro avg       0.84      0.78      0.80      9769
weighted avg       0.86      0.87      0.86      9769

Accuracy Score: 0.8671307196232982
Recall Score: 0.6134130713370355
Precision Score: 0.7851284855112083
ROC AUC Score:  0.7802525776717487
Threshold: [2 1 0]
```




```
pickle_out = open("model.pkl", "wb")
pickle.dump(gbc, pickle_out)
pickle_out.close()
```

Agenda

Background

Model

Flask API

Heroku

Flask API - CODE

```
import numpy as np
import pandas as pd
from flask import Flask, request, render_template
from sklearn import preprocessing
import pickle

app = Flask(__name__)
model = pickle.load(open('model.pkl', 'rb'))
cols=['age','workclass','education','marital-status','occupation','relationship','race','gender','capital-gain','capital-loss',
    'hours-per-week','native-country']

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

@app.route('/predict',methods=['POST'])
def predict():
  feature_list = request.form.to_dict()
  feature_list = list(feature_list.values())
  feature_list = list(map(int, feature_list))
  final_features = np.array(feature_list).reshape(1, 12)

  prediction = model.predict(final_features)
  output = int(prediction[0])
  if output == 1:
    text = ">50K"
  else:
    text = "<=50K"

  return render_template('index.html', prediction_text='Employee Income is {}'.format(text))


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

- The loaded model is in pickle format.
- The prediction has an additional in case it is negative.

Flask API - HTML

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML Deployment using Heroku</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>
</head>

<body>
  <div class="prediction">
    <h1>Income Prediction</h1>
    <form action="{{ url_for('predict')}}"method="post">
                                    <label for="age">Age</label>
                                    <input type="text" id="age" name="age" required="required">
                                    <br>
                                    <label for="workclass">Working Class</label>
                                    <select id="workclass" name="workclass" required="required">
                                                <option value="1">Federal-gov</option>
                                                <option value="2">Local-gov</option>
                                                <option value="3">Never-worked</option>
                                                <option value="4">Private</option>
                                                <option value="5">Self-emp-inc</option>
                                                <option value="6">Self-emp-not-inc</option>
                                                <option value="7">State-gov</option>
                                                <option value="8">Without-pay</option>
                                    </select>
                                    <br>
                                    <label for="education">Education</label>
                                    <select id="education" name="education" required="required">
                                                <option value="0">10th</option>
                                                <option value="1">11th</option>
                                                ...
```

- The HTML format used was provided by DATA GLACIER.

Agenda

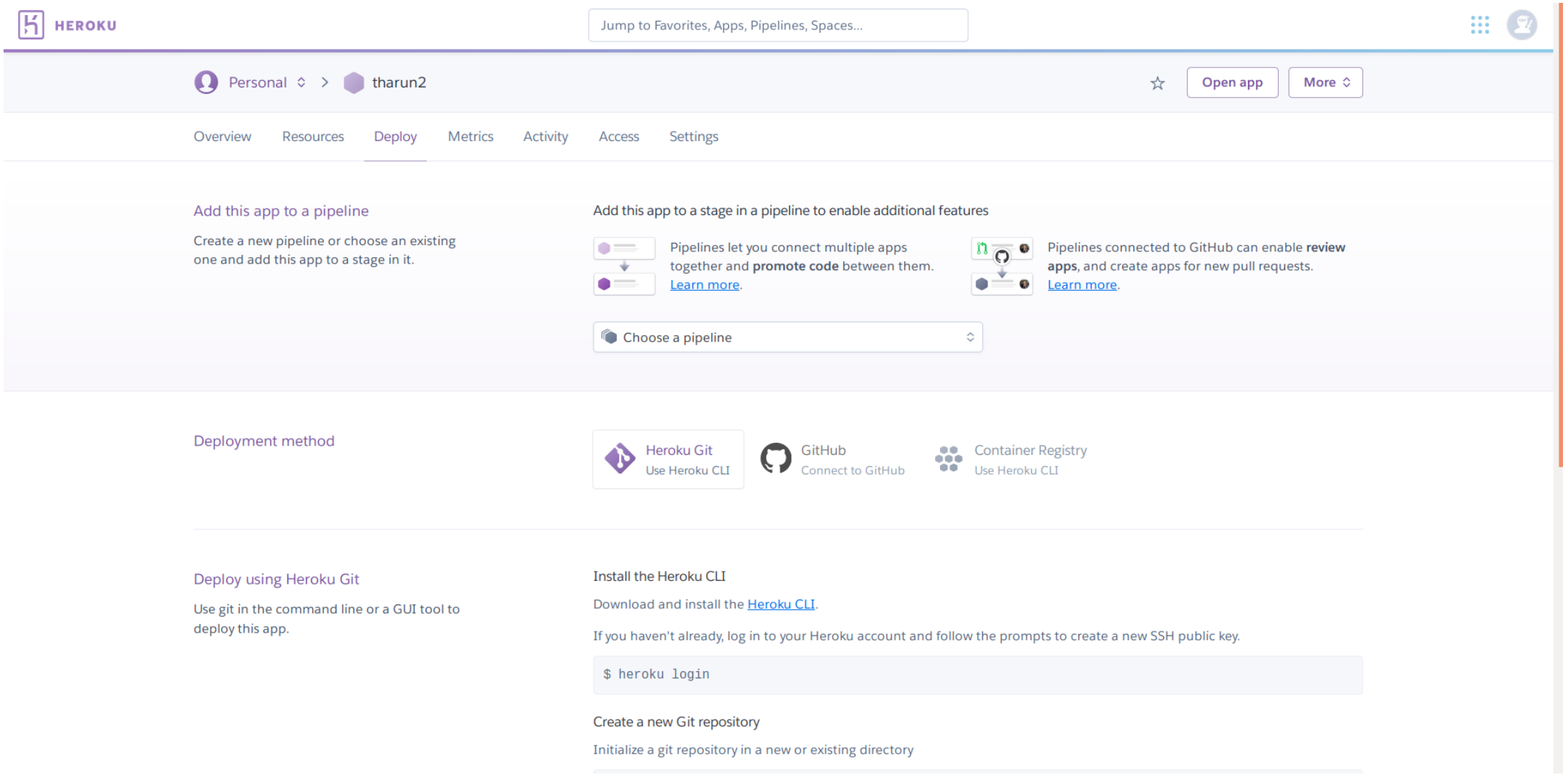
Background

Model

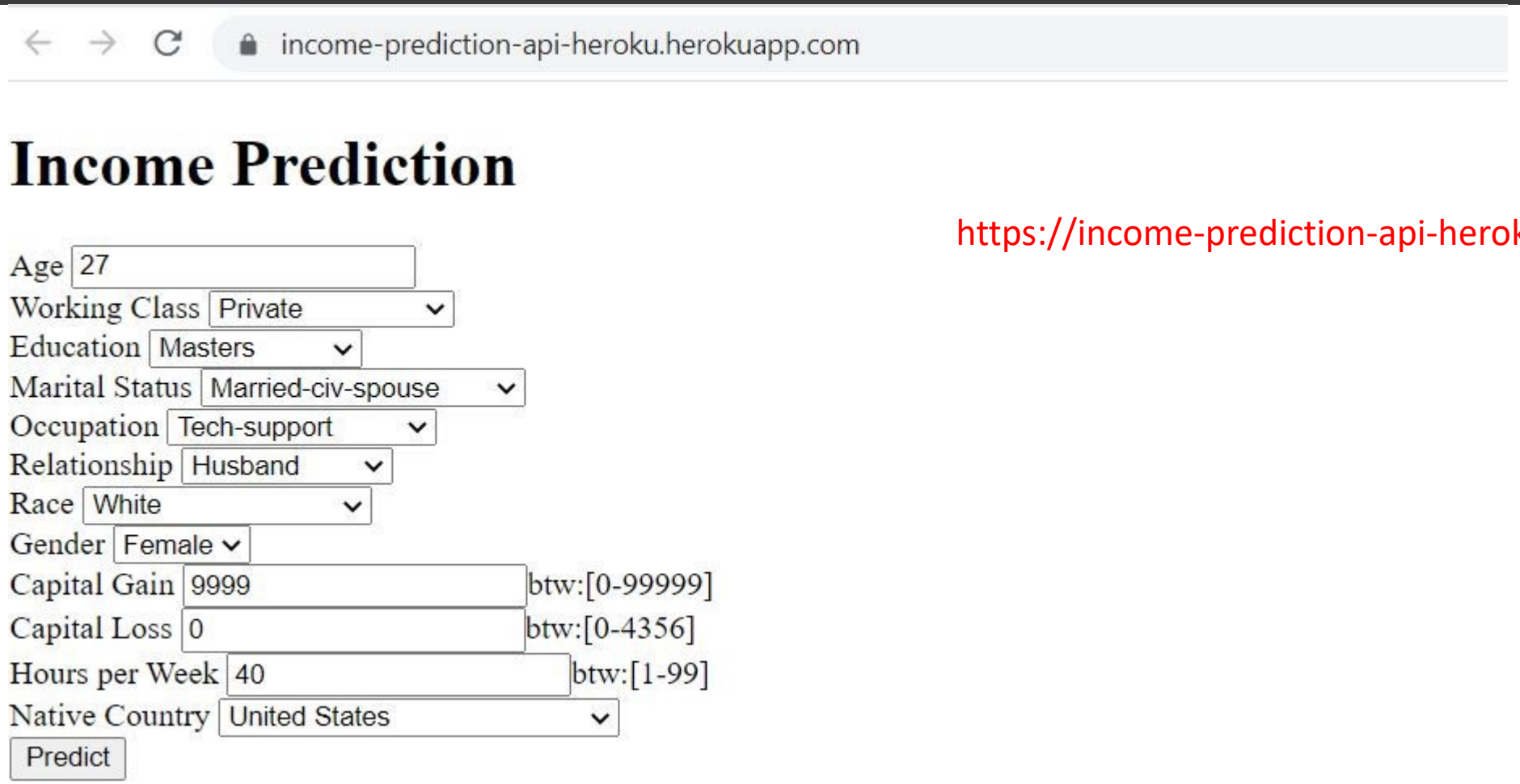
Flask API

Heroku

Heroku


HEROKU
Jump to Favorites, Apps, Pipelines, Spaces...
Personal
tharun2
Open app
More
Overview
Resources
Deploy
Metrics
Activity
Access
Settings
Add this app to a pipeline
Create a new pipeline or choose an existing one and add this app to a stage in it.
Add this app to a stage in a pipeline to enable additional features
Pipelines let you connect multiple apps together and promote code between them.
Learn more.
Pipelines connected to GitHub can enable review apps, and create apps for new pull requests.
Learn more.
Choose a pipeline
Deployment method
Heroku Git
Use Heroku CLI
GitHub
Connect to GitHub
Container Registry
Use Heroku CLI
Deploy using Heroku Git
Use git in the command line or a GUI tool to deploy this app.
Install the Heroku CLI
Download and install the Heroku CLI.
If you haven't already, log in to your Heroku account and follow the prompts to create a new SSH public key.
$ heroku login
Create a new Git repository
Initialize a git repository in a new or existing directory

Heroku



https://income-prediction-api-heroku.herokuapp.com/

- A web-app was developed where the machine learning model was implemented.

Thank You