

Министерство науки и высшего образования РФ
Федеральное государственное автономное образовательное
учреждение высшего образования
«Национальный исследовательский технологический университет
«МИСиС»

Институт Информационных технологий и компьютерных наук (ИТКН)
Кафедра Автоматизированных Систем Управления (АСУ)

ОТЧЁТ ПО УЧЕБНОЙ ПРАКТИКЕ

По теме: “Домашняя работа №3”

Выполнил:
студент группы БИВТ-20-1
Янушка А.В.
Проверил:
доцент кафедры АСУ
Громов С.В.

МОСКВА 2022

Задачи работы:

Написать скрипт, который будет автоматически подготавливать окружение и выполнять Jupyter сценарий с возможностью скачать вывод в виде html файла.

Используемые технологии:

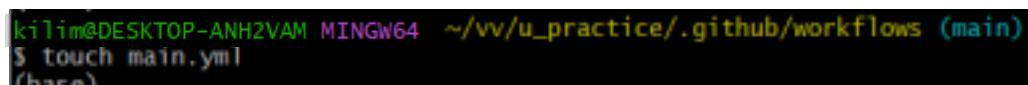
GitHub Actions - бесплатная для публичных репозиториях система непрерывной интеграции.

С помощью GitHub Actions можно делать много полезного:

- ✓ Запустить проверку кода линтером и тестами;
- ✓ Выполнить деплой проекта;
- ✓ Опубликовать новую версию пакета;
- ✓ Подключить оповещения в мессенджер о событиях в репозитории
- ✓ И др.

Описание решения задач и использованных подходов:

- 1) В моем локальном репозитории, при помощи Git Bash, в директории “.github/workflow” был создан файл “main.yml”:



```
kilim@DESKTOP-ANH2VAM MINGW64 ~/vv/u_practice/.github/workflows (main)
$ touch main.yml
(base)
```

- 2) Внутри файла была описана логика автоматической установки всего необходимого окружения и сценарий конвертирования Jupyter тетрадки в доступный для последующего скачивания формат:

```

name: Jupyter

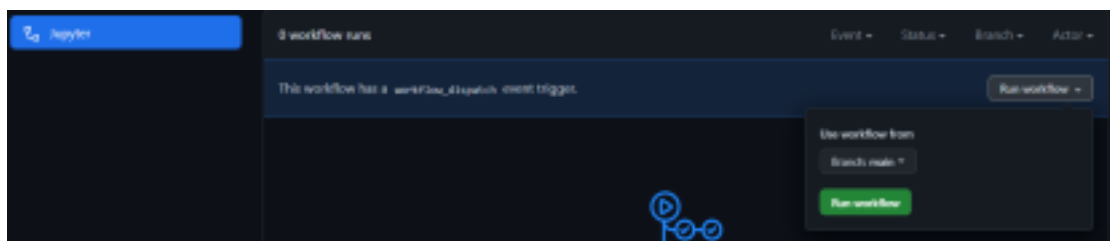
on:
  push:
    branches: [main]
  pull_request:

  workflow_dispatch:
jobs:
  build:
    runs-on: ubuntu-latest
    steps:|
      - name: Checkout
        uses: actions/checkout@v3
      - name: Setup Python
        uses: actions/setup-python@v4.0.0
        with:
          cache: pip
          python-version: '3.7'
      - name: Install Dependencies
        run: pip install -r requirements.txt
      - name: Execute Notebook
        run: jupyter-nbconvert --execute summer_un_pr.ipynb --to html
      - name: Upload a Build Artifact
        uses: actions/upload-artifact@v3.1.0
        with:
          name: Artifact
          path: summer_un_pr.html


```

3)

- 4) Когда мы создали сценарий и загрузили его, нужно перейти на вкладку Actions в GitHub и запустить скрипт:



- 5) Скачаем вывод (артефакт) Jupyter тетрадки. Он скачивается в формате zip архива.

Artifacts	
Produced during runtime	
Name	Size
 Artifact	575 KB



6) Далее проверяем конечный продукт:

```

in [30]:
# Генерация данных для логистической регрессии
import numpy as np
import random
from sklearn.datasets import load_iris, load_digits, load_wdbc
from sklearn.metrics import accuracy_score, confusion_matrix

# Генерация данных
X, y = load_digits(return_X_y=True)

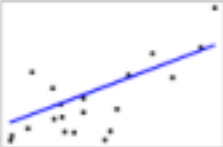
# Разделение данных на обучающую и тестовую
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Обучение модели
model = LogisticRegression()
model.fit(X_train, y_train)

# Проверка модели
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
confusion = confusion_matrix(y_test, y_pred)

print(f'Accuracy: {accuracy}')
print(f'Confusion Matrix: {confusion}')

```



```

in [31]:
from sklearn.metrics import roc_auc_score
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import numpy as np

# Генерация данных
X, y = load_digits(return_X_y=True)

# Разделение данных на обучающую и тестовую
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

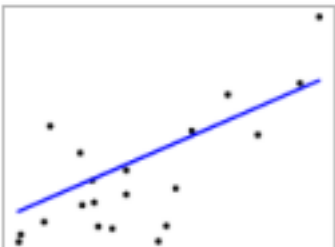
# Обучение модели
model = LogisticRegression()
model.fit(X_train, y_train)

# Проверка модели
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
confusion = confusion_matrix(y_test, y_pred)

# ROC Curve
roc_auc = roc_auc_score(y_test, model.predict_proba(X_test)[:, 1])

print(f'ROC AUC: {roc_auc}')

```



in [32]:

Результатом работы является артефакт, включающий в себя результат работы Jupyter notebook в формате html.

Заключение:

В ходе выполнения данной работы мною был написан скрипт, который преобразовывает определенный ipynb файл к html.