

Manual de Usuario y Desarrollador

Proyecto: Análisis de Regresión de Poisson

AUTOR: Juan Carlos Anquise Vargas

Introducción:

Este proyecto implementa una aplicación web desarrollada con Flask que permite realizar un análisis de regresión de Poisson utilizando datos generados automáticamente. La aplicación también permite descargar los datos generados en formato CSV o Excel.

Estructura del Proyecto

El proyecto está organizado en la siguiente estructura de carpetas:

project/

|

| — static/ # Recursos estáticos como imágenes

| | — img/

| | — UNA-Puno-2.png # Imagen de fondo

|

| — templates/ # Plantillas HTML para la aplicación Flask

| | — index.html # Página principal

|

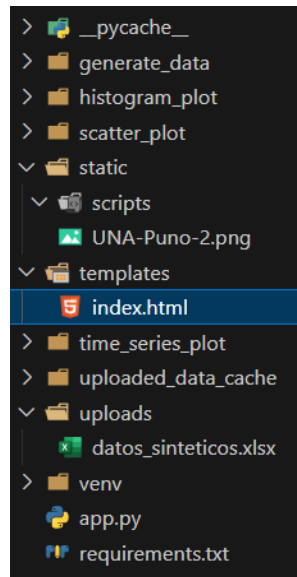
| — data/ # Carpeta para almacenar datos generados o subidos

|

| — app.py # Archivo principal que ejecuta el servidor Flask

| — requirements.txt # Dependencias del proyecto

| — README.md # Instrucciones y descripción del proyecto



Instalación:

Requisitos previos

1. Python 3.8 o superior.
2. Gestor de paquetes pip.
3. Entorno virtual (opcional) para aislar las dependencias del proyecto.

Pasos de instalación

1. Clona el repositorio del proyecto o descárgalo como archivo ZIP.
2. Navega a la carpeta del proyecto desde la terminal:
 - bash
 - cd project
3. Crea y activa un entorno virtual (opcional):
 - bash
 - python -m venv venv
 - En Windows: venv\Scripts\activate
4. Instala las dependencias del proyecto:
 - bash
 - pip install -r requirements.txt
5. Ejecuta la aplicación Flask:
 - bash
 - python app.py
6. Accede a la aplicación en tu navegador web utilizando la URL:
 - <http://127.0.0.1:5000/>

Uso de la Aplicación:

Inicio:

1. Al abrir la aplicación en tu navegador, verás una interfaz gráfica con los siguientes elementos:
 - Encabezados que identifican a la Universidad Nacional del Altiplano.
 - Un fondo personalizado con la imagen proporcionada.



2. Puedes interactuar con el formulario para generar datos o descargar datos existentes.

Generación de Datos

- Completa el formulario para indicar el número de muestras que deseas generar.
- Haz clic en el botón de generación.
- La aplicación generará un archivo `generated_data.csv` en la carpeta `data/` con las columnas predeterminadas.



#	Número de Empleados	Horas de Capacitación	Antigüedad Promedio	Presupuesto de Seguridad	Número de Supervisores	Número de Accidentes
1	152	14.68	4.38	82.81	3	0
2	485	11.74	5.96	35.97	11	1
3	398	35.99	8.82	61.14	10	0
4	320	28.03	7.62	67.39	11	0
5	156	31.24	5.79	23.72	6	0
6	121	10.62	9.95	68.60	3	0
7	238	39.10	3.81	33.64	5	0
8	70	34.97	5.80	25.20	13	0
9	152	16.37	6.76	95.91	13	0
10	171	15.45	7.93	97.25	8	0

Generar Datos Sintéticos

Número de Muestras:

6

Generar Datos

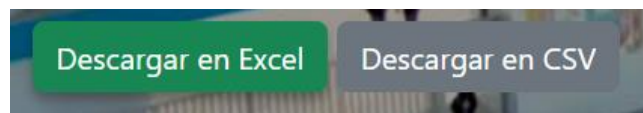
Datos Generados

#	Número de Empleados	Horas de Capacitación	Antigüedad Promedio	Presupuesto de Seguridad	Número de Supervisores	Número de Accidentes
1	152	27.96	11.20	34.67	11	1
2	485	14.68	2.27	44.34	13	1
3	398	14.68	14.61	61.98	13	0
4	320	11.74	12.82	54.56	4	0
5	156	35.99	4.76	43.30	13	0
6	121	28.03	4.36	68.95	8	0

Descargar en Excel Descargar en CSV

Descarga de Datos

- Haz clic en el botón de descarga.
- Selecciona el formato deseado: CSV o Excel.
- El archivo se descargará a tu computadora.



Cargar Archivos:

- Al cargar el archivo se visualiza el peso del archivo

Cargar Datos y Realizar Análisis

Selecciona un archivo Excel o CSV:

Seleccionar archivo generated_data.xlsx

Nombre del archivo: generated_data.xlsx

Tamaño del archivo: 5.50 KB

Tipo de archivo: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet

Subir y Analizar

- Analizamos y visualizamos los datos del archivo

Datos del Archivo Subido

#	n_empleados	horas_capacitacion	antiguedad_promedio	presupuesto_seguridad	n_supervisores	n_accidentes
1	152	14.6798356101	4.3842586281	82.8140769114	3	0
2	485	11.742508365	5.9551491585	35.9739025727	11	1
3	398	35.9852843732	8.8218336112	61.1387550731	10	0
4	320	28.0334503523	7.6152852423	67.393165509	11	0
5	156	31.2421773339	5.7859788226	23.7160330176	6	0
6	121	10.6175348289	9.9540876314	68.6035881521	3	0
7	238	39.0972955649	3.8134201885	33.641929895	5	0
8	70	34.973279224	5.797880431	25.2041274388	13	0
9	152	16.3701733203	6.7627039628	95.9108429803	13	0
10	171	15.4547490162	7.9289097948	97.250562646	8	0

- Visualizamos los resultados del archivo

Resultados del Análisis

Variable	Coficiente	Exp(Coficiente)	Cambio %	P-valor
const	-1.9455	0.1429	-85.71%	1.0000
n_empleados	0.0293	1.0297	2.97%	0.9998
horas_capacitacion	-0.5828	0.5584	-44.16%	0.9996
antiguedad_promedio	-0.7790	0.4589	-54.11%	0.9999
presupuesto_seguridad	-0.1638	0.8489	-15.11%	0.9997
n_supervisores	0.4641	1.5905	59.05%	0.9999

- Seleccionamos variables para generar los graficos



Archivos Clave

1. app.py

El archivo principal que ejecuta el servidor Flask. Contiene las rutas para:

- Cargar la página principal ('/').
- Generar datos en formato CSV ('/generate_data').
- Descargar datos en diferentes formatos ('/download_data').

Código destacado:

python

```
@app.route('/generate_data', methods=['POST'])
def generate_data():
    n_samples = int(request.form.get('n_samples', 100))
    data = pd.DataFrame({
        'n_empleados': [5] * n_samples,
        'horas_capacitacion': [10] * n_samples,
        'antiguedad_promedio': [2.5] * n_samples,
        'presupuesto_seguridad': [1000] * n_samples,
        'n_supervisores': [3] * n_samples,
        'n_accidentes': [1] * n_samples
    })
    data.to_csv('data/generated_data.csv', index=False)
    return data.to_json(orient='records')
'''
```

2. index.html

La plantilla HTML principal. Contiene la estructura de la interfaz del usuario y referencias a archivos CSS y JavaScript.

Cambiar el fondo

1. Sustituye la imagen `UNA-Puno-2.png` en `static/img/` por la imagen deseada.
2. Asegúrate de que el archivo tenga el mismo nombre o actualiza la referencia en `styles.css`:

css

```
background-image: url('/static/img/nueva-imagen.png')
```

Agregar nuevas columnas a los datos generados

1. Edita la función `generate_data` en `app.py`.
2. Agrega nuevas columnas al diccionario `data`.

Mantenimiento

Solución de Problemas

1. Error: “ModuleNotFoundError” al ejecutar `app.py`

- Verifica que las dependencias estén instaladas:

```
bash
```

```
pip install -r requirements.txt
```

2. Los estilos o scripts no se cargan

- Asegúrate de que la carpeta `static/` esté en la ubicación correcta y que los archivos tengan permisos adecuados.

3. El archivo generado no se descarga

- Verifica que la carpeta `data/` exista y tenga permisos de escritura.

Extensiones Futuras

1. Integración de modelos estadísticos: Implementar el análisis de regresión directamente en la aplicación.
2. Autenticación: Agregar una capa de autenticación para proteger el acceso.
3. Visualizaciones: Generar gráficas y reportes automáticos a partir de los datos generados.

CODIGO FUENTE EN GITHUB

Por el momento solamente analiza los datos sintéticos generados, una vez descargados y subir ese mismo archivo para visualizar los resultados del análisis

<https://github.com/JuanCarlosAnquise1007/codigo>