



Predicción de actividad humana en teléfonos inteligentes para la oportuna localización de sobrevivientes en catástrofes

Anexo 1: Manual de usuario

Andrés Camilo Caraballo Álvarez

Universidad Antonio Nariño
Facultad de Ingeniería de Sistemas
Bogotá, Colombia
2021

Contenido

	Pág.
1. Guía de usuario	4
2. Requerimientos del sistema.....	5
2.1. Requerimientos mínimos.....	5
2.2. Requerimientos recomendables.....	5
3. Instalación y descarga de recursos.....	6
3.1. Descarga Python en Windows:	6
3.2. Instalación Python en Windows.....	6
3.3. Instalación de bibliotecas	8
3.4. Descarga de archivos.....	9
4. Ejecución del proyecto y lectura de resultados.....	11
4.1. Ejecución del código.....	11
4.2. Lectura de resultados.....	13

Tabla de ilustraciones

Ilustración 1	Página de Python	6
Ilustración 2	Ventana emergente para la instalación de Python	6
Ilustración 3	Cuadro de instalación de Python.....	7
Ilustración 4	Cuadro de búsqueda de Windows.....	7
Ilustración 5	Ventana de Python	8
Ilustración 6	Cuadro de búsqueda de windows	8
Ilustración 7	Instrucción de Python para instalar numpy.....	8
Ilustración 8	Instrucción de Python para instalar scikit-learn	9
Ilustración 9	Instrucción de Python para instalar pandas.....	9
Ilustración 10	Instrucción de Python para instalar matplotlib	9
Ilustración 11	Documentos compartidos en Drive.....	9
Ilustración 12	Opciones para descargar los archivos de Drive	10
Ilustración 13	Cuadro de búsqueda de Windows.....	11
Ilustración 14	Comando para cambiar de carpeta	11
Ilustración 15	Comando para ejecutar el archivo transform.py	11
Ilustración 16	Comando para ejecutar el archivo preprocess.py	12
Ilustración 17	Comando para ejecutar el archivo svm.py	12
Ilustración 18	Comando para ejecutar el archivo Rna.py	12
Ilustración 19	Comando para ejecutar el archivo Random_Forest.py	12
Ilustración 20	Comando para ejecutar el archivo svm_with_rfe.py	12
Ilustración 21	Comando para ejecutar el archivo Rna_with_rfe.py	13
Ilustración 22	Comando para ejecutar el archivo rf_with_rfe.py	13
Ilustración 23	Ejemplo de documento con la elección de parámetros	13
Ilustración 24	Ejemplo de documento con los resultados de un modelo	13

1. Guía de usuario

El siguiente documento tiene como objetivo principal guiar al usuario por el funcionamiento y ampliar la comprensión del mismo, brindando información relevante de instalación y requerimientos para simular lo realizado en el proyecto de investigación.

De igual manera se dará un recorrido guiado del como ejecutar los códigos fuente de Python para que se realice de manera adecuada y en el orden adecuado, debido a que se debe realizar primero el pre procesamiento de los datos y luego el entrenamiento de los modelos presentados.

Por otro lado, se brindará el acceso a los documentos necesarios para realizar esta simulación, como lo son los archivos en formato .csv que se recolectaron a lo largo de la investigación, el código fuente de cada modelo y del proceso de pre procesamiento de los datos, también se adjuntaran los datos ya pre procesados y los resultados obtenidos por cada modelo en caso de únicamente querer verificar lo obtenido sin entrar a realizar el procedimiento completo.

2. Requerimientos del sistema

2.1. Requerimientos mínimos

- Sistema operativo: Windows 8 o versiones siguientes.
- Arquitectura del sistema: Windows -64-bit x86.
- Aplicación para poder abrir archivos .csv como Excel, Apache Open Office etc...
- No tener ninguna versión instalada de Python.
- Almacenamiento libre en disco mínimo de 6 MB para la instalación de Python y la descarga de documentos necesarios.
- Programa para descomprimir información por ejemplo winrar.

2.2. Requerimientos recomendables

No siempre los requerimientos mínimos nos brindarán el mejor desempeño en cuanto a lo que requiere el trabajo, es por esto que se recomiendan las siguientes adiciones o modificaciones a los requerimientos de la sección 2.1.

- Memoria RAM: 4GB o más para mejorar la velocidad en la que se ejecutan los procesos.
- Almacenamiento libre en disco mayor de 10MB para no ocupar todo el almacenamiento únicamente en el proyecto.
- Editor de texto diferente al bloc de notas, para mejorar la comprensión en los códigos fuente ejecutados, puede ser Sublime text, Notepad++, Visual Studio entre otros.

3. Instalación y descarga de recursos

3.1. Descarga Python en Windows:

- Abrir el navegador y diríjase al siguiente link:
<https://www.python.org/downloads/windows/>
- En la barra superior se encuentra la opción 'Downloads' pase el cursor por encima y se desplegará una lista de opciones.

Ilustración 1 Pagina de Python

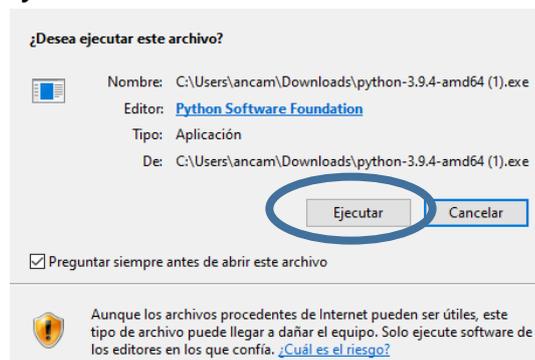


- Una vez allí se verá la opción que dice 'Download for windows' y un botón abajo que es la versión del Python que se descargará, de clic en ese botón para iniciar la descarga.

3.2. Instalación Python en Windows

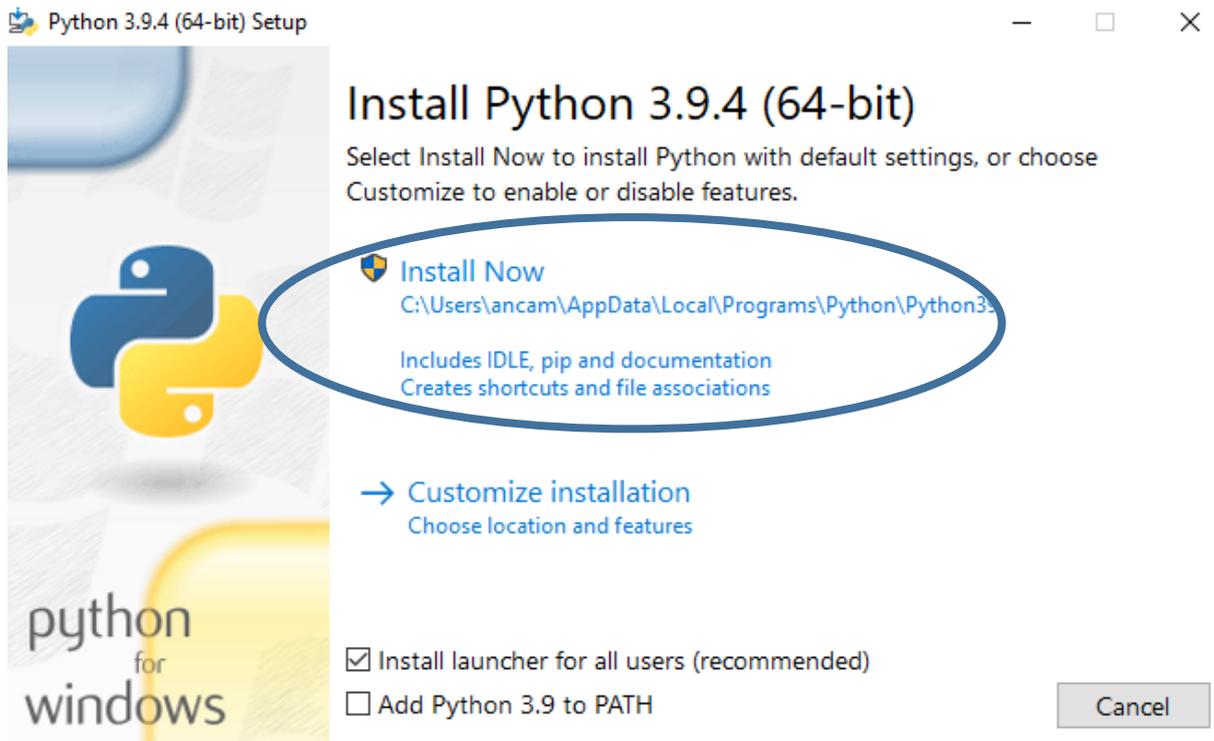
- Al ejecutar el instalador de Python aparecerá la siguiente pantalla en la cual escoge el botón ejecutar.

Ilustración 2 Ventana emergente para la instalación de Python



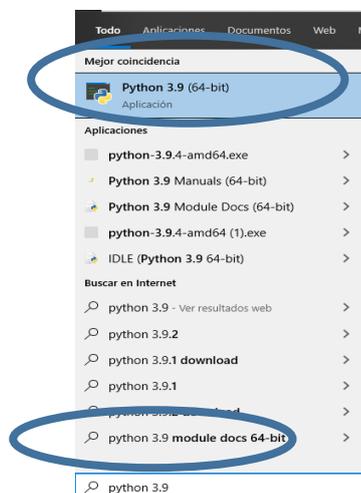
- b) Después de esto se desplegará otra ventana en la cual escoge la opción 'Install Now', (se debe contar con permisos de administrador o en su defecto tener la contraseña del usuario que tenga estos permisos dentro del equipo).

Ilustración 3 Cuadro de instalación de Python



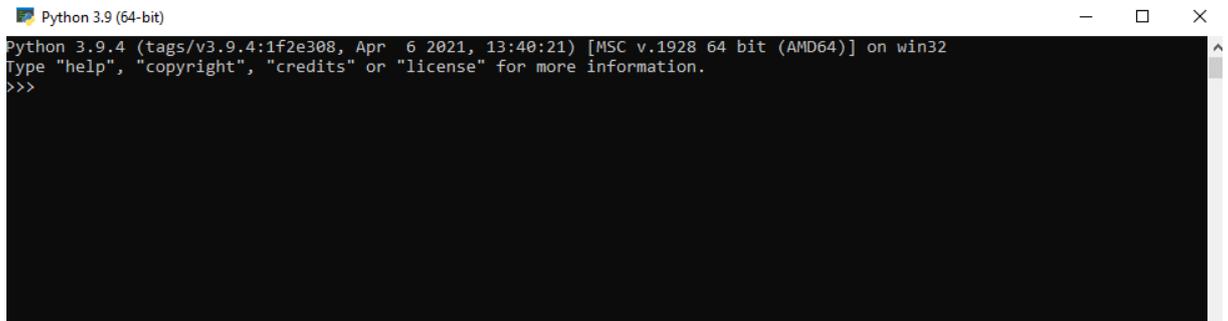
- c) Espere que la instalación se complete para finalizar el proceso, finalmente de clic en el botón 'Close' para concluir.
- d) Para verificar que la instalación fue exitosa en el buscador de Windows digite Python 3.9 y seleccione el primero.

Ilustración 4 Cuadro de búsqueda de Windows



- e) Debe aparecer una ventana similar a la siguiente, si es así es indicativo que la instalación fue correcta y puede cerrar la ventana.

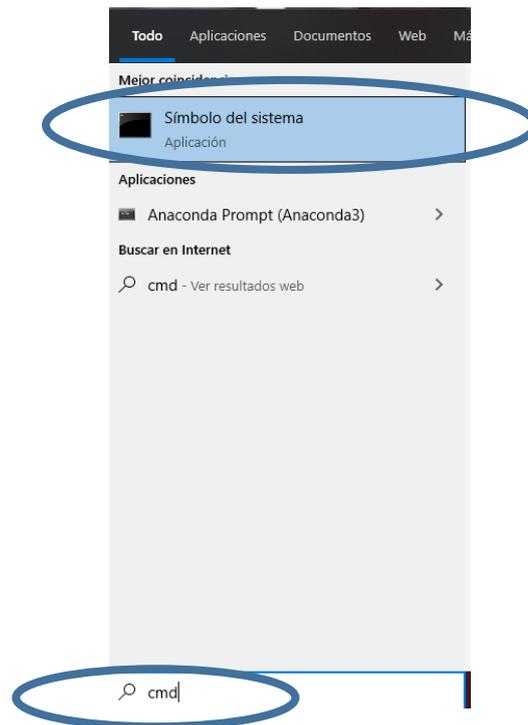
Ilustración 5 Ventana de Python



3.3. Instalación de bibliotecas

Para la instalación de las bibliotecas, en el cuadro de búsqueda digite cmd e ingrese a la primera opción.

Ilustración 6 Cuadro de búsqueda de windows



- a) Numpy

Para la instalación de numpy digite el comando “py -m pip install numpy” y oprima enter.

Ilustración 7 Instrucción de Python para instalar numpy

```
py -m pip install numpy
```

b) Scikit-learn

Para la instalación de scikit learn digite el comando “py -m pip install scikit-learn” y oprima enter.

Ilustración 8 Instrucción de Python para instalar scikit-learn

```
py -m pip install scikit-learn
```

c) Pandas

Para la instalación de pandas digite el comando “py -m pip install pandas” y oprima enter.

Ilustración 9 Instrucción de Python para instalar pandas

```
py -m pip install pandas
```

d) Matplotlib

Para la instalación de pandas digite el comando “py -m pip install matplotlib” y oprima enter.

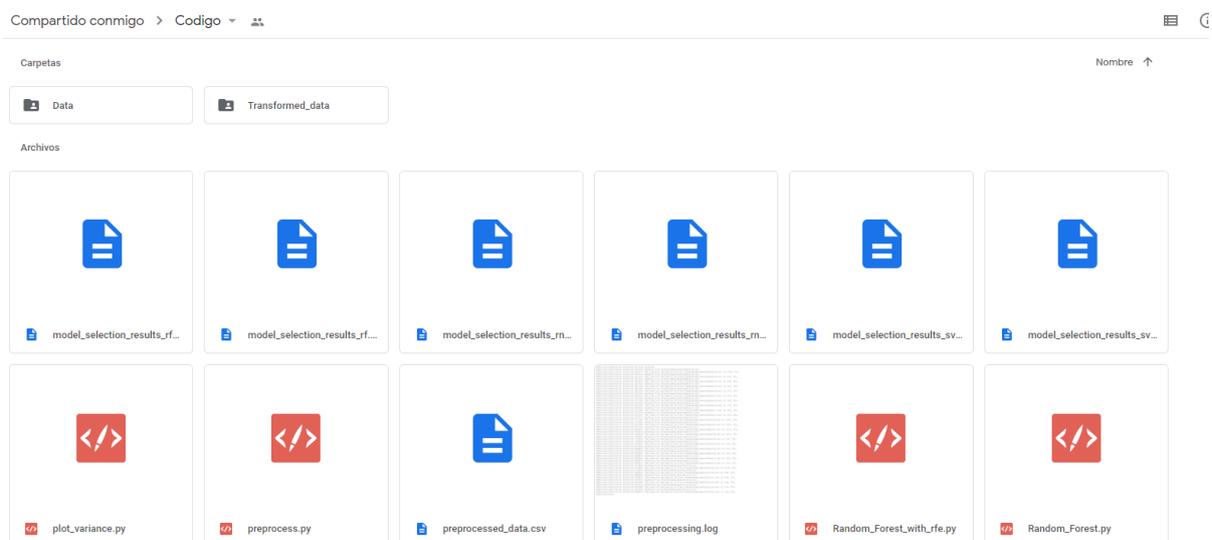
Ilustración 10 Instrucción de Python para instalar matplotlib

```
py -m pip install matplotlib
```

3.4. Descarga de archivos

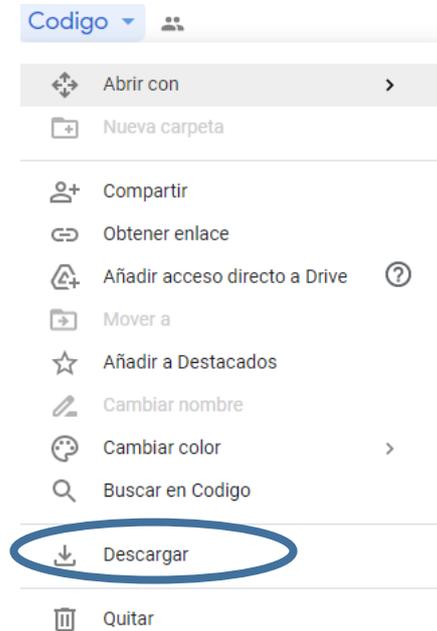
El código fuente de los modelos junto a la información necesaria para el entrenamiento se encuentran accediendo al siguiente link <https://drive.google.com/drive/folders/17Mf0XGVCEVAwxLJmJBDKTR7yP3Kq827V?usp=sharing>

Ilustración 11 Documentos compartidos en Drive



Aparecerá algo similar a la imagen anterior, allí le da clic a la opción código, en la lista desplegable encontrará la opción descargar.

Ilustración 12 Opciones para descargar los archivos de Drive



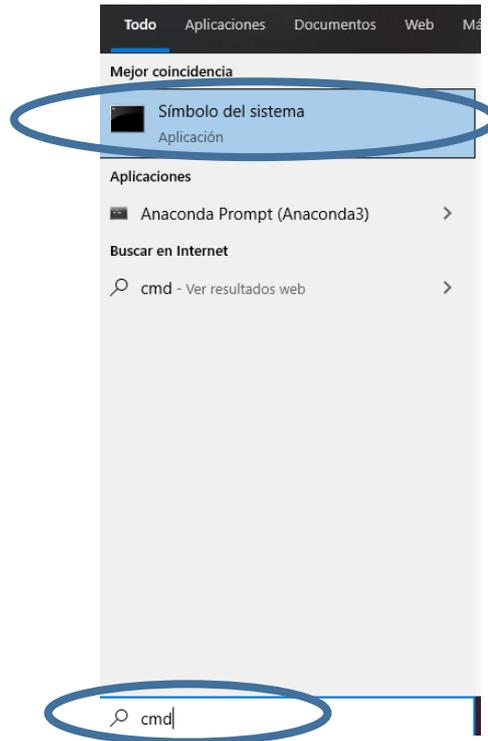
Se descargará un archivo .zip el cual deberá descomprimir en alguna dirección cómoda para trabajarlo posteriormente.

4. Ejecución del proyecto y lectura de resultados

4.1. Ejecución del código.

- a) Una vez se haya situado y descomprimido la carpeta en una dirección de almacenamiento cómoda, nos dirigiremos a la sección de búsqueda del entorno gráfico de Windows y buscamos 'cmd', ingresamos a la primera opción.

Ilustración 13 Cuadro de búsqueda de Windows



- b) Se dirige a la dirección en la cual descomprimió el archivo digitando lo siguiente 'cd Desktop\Codigo' (esto en caso que la carpeta se encuentre en el escritorio de lo contrario después del 'cd' se tendrá que especificar la dirección donde se encuentre).

Ilustración 14 Comando para cambiar de carpeta

```
cd Desktop\Codigo
```

- c) Lo primero es realizar la transformación de la información, es por esto que empezará ejecutando el archivo 'transform.py' digitando el comando 'py transform.py Data Transformed_data 2.5 transformation.log'.

Ilustración 15 Comando para ejecutar el archivo transform.py

```
py transform.py Data Transformed_data 2.5 transformation.log
```

El fragmento 'transform.py' se refiere al archivo que estamos ejecutando, 'Data' es la carpeta donde se encuentran los datos que vamos a transformar,

'Transformed_data' es la carpeta donde se situaran los nuevos archivos transformados, 'transformation.log' será un archivo donde se guardará la información detallada de lo que hace la ejecución del comando.

- d) Una vez hecho esto se procede a realizar el pre procesamiento de la información la cual se ejecutará con el comando 'py preprocess.py Transformed_data preprocessed_data.csv preprocessing.log'.

Ilustración 16 Comando para ejecutar el archivo preprocess.py

```
py preprocess.py Transformed_data preprocessed_data.csv preprocessing.log
```

- e) Con la información ya lista para entrar a los modelos de entrenamiento ejecutamos el primero que es máquinas de soporte de vectores seguirá el comando 'py svm.py preprocessed_data.csv svm_performance_metrics.csv model_selection_results_svm.csv 6 svm.log'.

Ilustración 17 Comando para ejecutar el archivo svm.py

```
py svm.py preprocessed_data.csv svm_performance_metrics.csv model_selection_results_svm.csv 6 svm.log
```

- f) El segundo modelo será redes neuronales artificiales que sigue el comando 'py Rna.py preprocessed_data.csv rna_performance_metrics.csv model_selection_results_rna.csv 6 rna.log'.

Ilustración 18 Comando para ejecutar el archivo Rna.py

```
py Rna.py preprocessed_data.csv rna_performance_metrics.csv model_selection_results_rna.csv 6 rna.log
```

- g) El tercer modelo es el de bosques aleatorios cuyo comando es 'py Random_Forest.py preprocessed_data.csv rf_performance_metrics.csv model_selection_results_rf.csv 6 rf.log'.

Ilustración 19 Comando para ejecutar el archivo Random_Forest.py

```
py Random_Forest.py preprocessed_data.csv rf_performance_metrics.csv model_selection_results_rf.csv 6 rf.log
```

- h) A partir de aquí se incluyeron los mismos modelos, pero con diferente método de selección de variables, este es el de máquinas de soporte de vectores con RFE. 'py svm_with_rfe.py preprocessed_data.csv svm_with_rfe_performance_metrics.csv model_selection_results_svm_with_rfe.csv 6 svm_with_rfe.log'

Ilustración 20 Comando para ejecutar el archivo svm_with_rfe.py

```
py svm_with_rfe.py preprocessed_data.csv svm_with_rfe_performance_metrics.csv model_selection_results_svm_with_rfe.csv 6 svm_with_rfe.log
```

- i) El segundo modelo corresponde a redes neuronales artificiales con RFE siguiendo el comando 'py Rna_with_rfe.py preprocessed_data.csv rna_with_rfe_performance_metrics.csv model_selection_results_rna_with_rfe.csv 6 rna_with_rfe.log'

Ilustración 21 Comando para ejecutar el archivo Rna_with_rfe.py

```
C:\Users\ancam\Desktop\Codigo>py Rna_with_rfe.py preprocessed_data.csv rna_with_rfe_performance_metrics.csv model_selection_results_rna_with_rfe.csv 6 rna_with_rfe.log
model_selection_results_rna_with_rfe.csv 6 rna_with_rfe.log'
```

- j) Por último, se evalúa bosques aleatorios con RFE con el comando 'py Random_Forest_with_rfe.py preprocessed_data.csv rf_with_rfe_performance_metrics.csv model_selection_results_rf_with_rfe.csv 6 rf_with_rfe.log'

Ilustración 22 Comando para ejecutar el archivo rf_with_rfe.py

```
C:\Users\ancam\Desktop\Codigo>py Rna_with_rfe.py preprocessed_data.csv rna_with_rfe_performance_metrics.csv model_selection_results_rna_with_rfe.csv 6 rna_with_rfe.log
```

4.2. Lectura de resultados

Dentro de la carpeta que estamos trabajando se encuentran varios archivos dentro de los cuales hay unos con extensión .csv que veremos para evaluar los resultados.

El primer tipo de archivos cuyo nombre sigue el formato 'model_selection_results_Nombre del modelo', estos archivos conservan en su interior la selección de parámetros de cada modelo, el archivo se ve como en la imagen siguiente.

Ilustración 23 Ejemplo de documento con la elección de parámetros

activation,logistic
alpha,0.001
hidden_layer_sizes,"(10,)"
learning_rate,adaptive
solver,lbfgs

El segundo tipo de archivos siguen el formato de nombre 'Nombre del modelo_performance_metrics' en el cual se encuentra toda la información del desempeño que dado modelo consiguió con la combinación de parámetros anterior mencionada, estos archivos lucen como la siguiente imagen.

Ilustración 24 Ejemplo de documento con los resultados de un modelo

	ROC AUC	F1	Precision	Recall	Accuracy	Balanced accuracy	Average precision
k = 10	0.9820421245421246	0.9478553048053604	0.9443174603174602	0.95	0.9373265494912119	0.9273095238095237	0.9774843101607151
k = 5	0.9753678769819374	0.9424654488960329	0.9469932342290832	0.9493966817496229	0.9416783543712015	0.9356362432402605	0.975790205005333