

# Instalación de Anaconda en Ubuntu Linux: Una guía paso a paso para instalar y configurar Anaconda en Ubuntu Linux y aprovechar al máximo su plataforma de gestión de paquetes y entornos virtuales.

Edison Achalma

Escuela Profesional de Economía, Universidad Nacional de San Cristóbal de Huamanga

Primer parrafo de abstrac

*Palabras Claves:* keyword1, keyword2

## Tabla de contenidos

<b>Introduction</b>	<b>1</b>
<b>Introducción</b>	<b>1</b>
<b>¿Qué es Anaconda?</b>	<b>1</b>
<b>Pasos para la instalación de Anaconda en Ubuntu Linux</b>	<b>2</b>
Descargar el instalador de Anaconda . . . . .	2
<b>Verificación de la instalación de Anaconda</b>	<b>4</b>
<b>Actualización de Anaconda en Ubuntu Linux</b>	<b>4</b>
<b>Desinstalación de Anaconda en Ubuntu Linux</b>	<b>5</b>
<b>Comandos básicos</b>	<b>5</b>
<b>Publicaciones Similares</b>	<b>6</b>

## Instalación de Anaconda en Ubuntu Linux

### Introducción

La instalación de Anaconda en Ubuntu Linux es un paso crucial para aquellos que desean aprovechar al máximo el desarrollo en Python en este sistema operativo. Anaconda, una plataforma de gestión de paquetes y entornos virtuales, ofrece una amplia gama de herramientas y bibliotecas que facilitan el trabajo con Python, así como con otros lenguajes de programación populares. Al instalar Anaconda en Ubuntu Linux, los desarrolladores obtienen una solución integral que simplifica el proceso de configuración y administración de su entorno de desarrollo.

La importancia de la instalación de Anaconda en Ubuntu Linux radica en su capacidad para proporcionar un flujo de trabajo fluido y eficiente. Al ofrecer una amplia selección de paquetes precompilados, Anaconda elimina la necesidad de buscar e instalar manualmente cada biblioteca requerida. Esto ahorra tiempo y evita posibles conflictos de dependencia, permitiendo a los desarrolladores centrarse en la programación en lugar de la configuración del entorno.

Además de su conveniencia, Anaconda también ofrece beneficios significativos en términos de gestión de entornos virtuales. Con la ayuda de la herramienta “conda”, los desarrolladores pueden crear y administrar fácilmente entornos aislados para proyectos específicos. Esto permite mantener diferentes versiones de paquetes y bibliotecas para cada proyecto, evitando conflictos y facilitando la replicación del entorno de desarrollo en diferentes sistemas.

¡Sumérgete en el mundo de Anaconda y descubre cómo simplificar y mejorar tu experiencia de desarrollo en Python en Ubuntu Linux!

### ¿Qué es Anaconda?

Anaconda es una plataforma de gestión de paquetes y entornos virtuales diseñada para facilitar el desarrollo en Python y otros lenguajes de programación populares. Imagina que tienes una caja de herramientas repleta de todo lo que necesitas para construir aplicaciones, analizar datos o desarrollar

---

 Edison Achalma

El autor no tiene conflictos de interés que revelar. Los roles de autor se clasificaron utilizando la taxonomía de roles de colaborador (CRediT; <https://credit.niso.org/>) de la siguiente manera: Edison Achalma: conceptualización, redacción

La correspondencia relativa a este artículo debe dirigirse a Edison Achalma, Email: [elmer.achalma.09@unsch.edu.pe](mailto:elmer.achalma.09@unsch.edu.pe)

proyectos de aprendizaje automático. Esa es Anaconda: una caja de herramientas poderosa y completa que te brinda todo lo que necesitas para trabajar con Python de manera eficiente.

Una de las principales ventajas de Anaconda es su capacidad para gestionar paquetes. En lugar de buscar y descargar manualmente cada biblioteca que necesitas, Anaconda te ofrece una amplia selección de paquetes precompilados y listos para usar. Esto significa que no tienes que preocuparte por las dependencias o por perder tiempo en configuraciones complicadas. Con Anaconda, puedes comenzar a trabajar en tu proyecto de inmediato.

Otra gran ventaja de Anaconda es su capacidad para crear entornos virtuales. ¿Qué significa esto? Básicamente, puedes aislar tus proyectos en entornos separados, cada uno con su propia configuración de paquetes y versiones. Esto es especialmente útil cuando trabajas en varios proyectos al mismo tiempo o cuando necesitas mantener diferentes versiones de bibliotecas para distintas aplicaciones. Anaconda hace que la gestión de entornos sea sencilla y te permite alternar entre ellos con facilidad.

## Pasos para la instalación de Anaconda en Ubuntu Linux

### Descargar el instalador de Anaconda

**Paso 1:** Accede al sitio web oficial de Anaconda. Puedes hacerlo abriendo tu navegador web favorito y dirigiéndote a [Anaconda.com](https://anaconda.com). Una vez allí, busca la sección de descargas.

**Paso 2:** En la sección de descargas, encontrarás diferentes opciones de Anaconda según tu sistema operativo. Como estamos trabajando en Ubuntu Linux, asegúrate de seleccionar la versión adecuada para Linux.

**Paso 3:** Una vez que hayas seleccionado la versión de Linux, verás dos opciones: Anaconda Individual Edition y Anaconda Enterprise. Para la mayoría de los casos, Anaconda Individual Edition es la opción recomendada, ya que es una versión gratuita y completa de Anaconda. Haz clic en el botón de descarga correspondiente a la edición que desees instalar.

**Paso 4:** La descarga comenzará y, dependiendo de la velocidad de tu conexión a Internet, podría llevar algunos minutos. Asegúrate de esperar hasta que la descarga se complete antes de pasar al siguiente paso.

Ahora que has descargado el instalador de Anaconda en tu sistema Ubuntu Linux, es hora de abrir la terminal y ejecutar algunos comandos para continuar con el proceso de instalación. ¡No te preocupes, te guiaré paso a paso!

**Paso 5:** Abre la terminal. Puedes hacerlo de diferentes maneras, pero una forma común es presionar las teclas Ctrl + Alt + T al mismo tiempo. Esto abrirá una nueva ventana de terminal en tu pantalla.

**Paso 6:** Una vez que tengas la terminal abierta, es importante asegurarte de que estás ubicado en el directorio correcto. Para hacerlo, ejecuta el comando “pwd” y presiona Enter. Esto te mostrará la ruta actual en la que te encuentras. Asegúrate

de estar en el lugar adecuado antes de continuar.

**Paso 7:** Ahora, necesitamos navegar a la ubicación donde se encuentra el instalador de Anaconda que descargaste. Por lo general, se guarda en la carpeta “Downloads” (Descargas). Para acceder a esta carpeta, ejecuta el siguiente comando:

```
cd Downloads/
```

Presiona Enter después de escribir el comando y verás que la ruta de la terminal cambia al directorio “Downloads”.

**Paso 8:** ¡Estamos casi listos para ejecutar el instalador de Anaconda! Ahora, debes ejecutar el archivo de instalación. Asegúrate de que el nombre del archivo coincida con el que descargaste. Para ejecutarlo, utiliza el siguiente comando:

```
bash nombre-del-archivo.sh
```

Reemplaza “nombre-del-archivo” con el nombre exacto del archivo de instalación que descargaste. Por ejemplo, si el archivo se llama “Anaconda3-2021.05-Linux-x86\_64.sh”, el comando sería:

```
bash Anaconda3-2021.05-Linux-x86_64.sh
```

Presiona Enter y comenzará el proceso de instalación de Anaconda. Sigue las instrucciones que aparecerán en la terminal y asegúrate de leer y aceptar los términos de licencia.

**Paso 9:** ¡Genial! Ahora que has ejecutado el instalador de Anaconda en Ubuntu Linux, solo te queda seguir las instrucciones de instalación que aparecerán en la terminal. No te preocupes, te guiaré a través de este paso.

Después de ejecutar el comando para iniciar la instalación, verás que aparecerá una serie de instrucciones en la terminal. Lee cuidadosamente cada paso y asegúrate de seguirlos correctamente.

Por lo general, las instrucciones te pedirán que revises y aceptes los términos de licencia. Para hacerlo, simplemente lee los términos y condiciones que se muestran en la terminal y, cuando se te solicite, presiona Enter para avanzar y aceptar los términos.

A continuación, se te pedirá que especifiques la ubicación de la instalación de Anaconda. Por defecto, suele ser en tu directorio de inicio, pero puedes elegir una ubicación diferente si lo deseas. Si no estás seguro, te recomendaría que aceptes la ubicación predeterminada presionando Enter.

Luego, la instalación te preguntará si deseas agregar Anaconda a tu variable de entorno PATH. Esto es útil para que puedas acceder a los comandos de Anaconda desde cualquier ubicación en tu sistema. Te sugeriría que respondas “yes” (sí) y presiones Enter.

Cuando la carpeta de anaconda ya existe, actualizamos programas existente.

```
bash Anaconda3-2021.11-Linux-x86_64.sh -u
```

Después de eso, la instalación continuará y verás una barra de progreso que indica el avance del proceso. Puede llevar algún tiempo, así que ten paciencia.

Una vez que la instalación se complete con éxito, verás un mensaje indicando que Anaconda se ha instalado correctamente. ¡Felicidades! Has instalado Anaconda en tu sistema Ubuntu Linux.

Ahora puedes cerrar la terminal y comenzar a disfrutar de las ventajas de utilizar Anaconda en tu sistema. Recuerda que Anaconda te ofrece un entorno de gestión de paquetes y entornos virtuales que te facilitará el trabajo con Python y otras herramientas de ciencia de datos.

**Paso 10:** Una vez que hayas instalado Anaconda en tu sistema Ubuntu Linux, es importante añadirlo al PATH del sistema para que puedas acceder a los comandos de Anaconda desde cualquier ubicación en tu sistema. Te guiaré a través de este paso.

1. Abre la terminal en tu sistema Ubuntu Linux. Puedes hacerlo buscando “Terminal” en el menú de aplicaciones o usando el atajo de teclado Ctrl+Alt+T.
2. En la terminal, escribe el siguiente comando para abrir el archivo de configuración de inicio de sesión:

```
nano ~/.bashrc
```

3. Esto abrirá el archivo `.bashrc` en el editor de texto nano. Ahora, desplázate hasta el final del archivo.
4. En la última línea del archivo, añade el siguiente comando para agregar Anaconda al PATH:

```
export PATH="/ruta/donde/instalaste/anaconda/bin:
```

Recuerda reemplazar “ruta/donde/instalaste/anaconda” con la ubicación real donde instalaste Anaconda. Si elegiste la ubicación predeterminada, el comando será:

```
export PATH="$HOME/anaconda/bin:$PATH"
```

5. Después de añadir el comando, guarda los cambios en el archivo `~/.bashrc`. En nano, puedes hacerlo presionando Ctrl+O, luego presiona Enter para confirmar y finalmente presiona Ctrl+X para salir del editor nano.
6. Ahora, para que los cambios tengan efecto, actualiza el archivo de configuración de inicio de sesión con el siguiente comando:

```
source ~/.bashrc
```

¡Y eso es todo! Has añadido correctamente Anaconda al PATH del sistema en Ubuntu Linux. Ahora podrás acceder a los comandos de Anaconda desde cualquier ubicación en tu sistema.

**Algunos problemas:** A veces, durante la instalación de Anaconda, puede ocurrir que no se pueda responder con “yes” al prompt de inicialización de Anaconda en el archivo `.bashrc`. Aquí te proporcionaré una solución detallada para abordar este problema:

1. Después de que la instalación de Anaconda se complete y se muestre el prompt para inicializar Anaconda en el archivo `.bashrc`, si no puedes responder con “yes” y el proceso se cierra, no te preocupes.
2. Abre la terminal en tu sistema Ubuntu Linux. Puedes hacerlo buscando “Terminal” en el menú de aplicaciones o usando el atajo de teclado Ctrl+Alt+T.
3. En la terminal, ejecuta el siguiente comando para abrir el archivo `.bashrc` en un editor de texto:

```
sudo gedit ~/.bashrc
```

Esto abrirá el archivo `.bashrc` con privilegios de administrador para poder editarlo.

4. Desplázate hasta el final del archivo `.bashrc` y agrega el siguiente bloque de código:

```
# >>> conda initialize >>>
!! Contents within this block are managed by 'conda
__conda_setup="$(('/home/achalmaubuntu/anaconda3/b
if [ $? -eq 0 ]; then
    eval "$__conda_setup"
else
    if [ -f "/home/achalmaubuntu/anaconda3/etc/pro
        . "/home/achalmaubuntu/anaconda3/etc/prof
    else
        export PATH="/home/achalmaubuntu/anaconda
    fi
fi
unset __conda_setup
# <<< conda initialize <<<
```

Recuerda cambiar el achalmaubuntu con su usuario

5. Guarda los cambios en el archivo `.bashrc`.
6. Luego, en la terminal, ejecuta el siguiente comando para actualizar las variables de entorno con los cambios realizados en el archivo `.bashrc`:

```
source ~/.bashrc
```

Recuerda reiniciar la terminal o abrir una nueva ventana de terminal para que los cambios surtan efecto.

¡Y eso es todo! Has solucionado el problema de inicialización de Anaconda en el archivo `.bashrc` y ahora puedes utilizar Anaconda y sus comandos sin problemas en tu sistema Ubuntu Linux.

### Verificación de la instalación de Anaconda

Una vez que hayas completado la instalación de Anaconda en tu sistema Ubuntu Linux, es importante verificar que todo haya sido instalado correctamente. Aquí te mostraré cómo comprobarlo y también cómo verificar la versión de Anaconda y los paquetes instalados.

#### Comprobar la instalación de Anaconda:

1. Abre la terminal en tu sistema Ubuntu Linux.
2. En la terminal, ingresa el siguiente comando:

```
conda --version
```

Esto mostrará la versión de Anaconda instalada en tu sistema. Si se muestra la versión correctamente, significa que Anaconda está instalado correctamente.

#### Verificar la versión de Anaconda:

1. En la terminal, ingresa el siguiente comando:

```
conda list anaconda$
```

Esto mostrará una lista de los paquetes instalados con el nombre “anaconda” en su versión. Si se muestra la lista de paquetes correctamente, significa que la instalación de Anaconda incluyendo los paquetes se ha realizado correctamente.

2. Para verificar la versión de un paquete específico, puedes ingresar el siguiente comando en la terminal, reemplazando `<nombre_paquete>` con el nombre del paquete que deseas verificar:

```
conda list <nombre_paquete>
```

Esto mostrará la versión del paquete específico instalado en tu sistema.

¡Listo! Ahora puedes asegurarte de que Anaconda esté correctamente instalado y verificar las versiones de Anaconda y los paquetes instalados en tu sistema Ubuntu Linux. Esto te ayudará a garantizar un funcionamiento adecuado de Anaconda y sus herramientas para el desarrollo de proyectos.

## Actualización de Anaconda en Ubuntu Linux

Mantener tu instalación de Anaconda actualizada es crucial para aprovechar las últimas mejoras, características y correcciones de errores. Afortunadamente, actualizar Anaconda en Ubuntu Linux es un proceso sencillo. A continuación, te mostraré los pasos para actualizar Anaconda a la última versión utilizando los comandos de actualización de Conda y paquetes.

#### Pasos para actualizar Anaconda:

1. Abre la terminal en tu sistema Ubuntu Linux.
2. En la terminal, ingresa el siguiente comando para actualizar Conda, el administrador de paquetes de Anaconda:

```
conda update conda
```

Este comando actualizará Conda a la última versión disponible.

3. A continuación, puedes utilizar el siguiente comando para actualizar todos los paquetes instalados en tu entorno de Anaconda:

```
conda update --all
```

Este comando buscará actualizaciones para todos los paquetes instalados y los actualizará a las últimas versiones disponibles.

4. Si deseas actualizar un paquete específico, puedes utilizar el siguiente comando, reemplazando `<nombre_paquete>` con el nombre del paquete que deseas actualizar:

```
conda update <nombre_paquete>
```

Esto actualizará el paquete especificado a la última versión disponible.

Recuerda que antes de realizar la actualización, es recomendable crear un entorno virtual y respaldar tus proyectos para evitar cualquier conflicto o pérdida de datos.

¡Y eso es todo! Con estos pasos, podrás mantener tu instalación de Anaconda actualizada y disfrutar de las últimas mejoras en Ubuntu Linux. Asegúrate de realizar actualizaciones periódicas para aprovechar al máximo las capacidades de Anaconda y sus paquetes.

## Desinstalación de Anaconda en Ubuntu Linux

Si en algún momento deseas eliminar completamente Anaconda de tu sistema Ubuntu Linux, ya sea para realizar una nueva instalación o por cualquier otro motivo, puedes seguir estos pasos para desinstalarlo por completo.

### Proceso para eliminar completamente Anaconda:

1. Abre la terminal en tu sistema Ubuntu Linux.
2. En la terminal, ingresa el siguiente comando para desactivar cualquier configuración de Anaconda que esté activa en tu entorno actual:

```
conda deactivate
```

Esto asegurará que no haya entornos virtuales activos relacionados con Anaconda.

3. A continuación, puedes utilizar el siguiente comando para desinstalar Anaconda y eliminar todos los archivos relacionados:

```
anaconda-clean --yes
```

Este comando eliminará Anaconda de tu sistema y eliminará los archivos y directorios asociados.

4. Una vez completado el proceso de desinstalación, puedes verificar que Anaconda se haya eliminado por completo. Puedes utilizar el siguiente comando para verificar si el directorio de Anaconda ya no existe:

```
ls ~/anaconda3
```

Si el directorio no se encuentra, significa que Anaconda ha sido eliminado correctamente.

Recuerda que este proceso eliminará permanentemente Anaconda de tu sistema, incluyendo todos los paquetes y entornos virtuales asociados. Asegúrate de realizar una copia de seguridad de tus proyectos y datos importantes antes de proceder con la desinstalación.

¡Y eso es todo! Siguiendo estos pasos, podrás eliminar completamente Anaconda de tu sistema Ubuntu Linux. Si en el futuro decides volver a instalarlo, podrás seguir los pasos que hemos cubierto anteriormente en este blog.

### Comandos básicos

1. `conda deactivate`: Este comando se utiliza para desactivar el entorno activo de Conda. Cuando trabajas con entornos virtuales en Anaconda, puedes utilizar este comando para volver al entorno base o desactivar cualquier entorno virtual activo.

2. `conda activate`: Con este comando, puedes activar un entorno virtual específico en Anaconda. Puedes crear entornos virtuales personalizados con diferentes versiones de Python y paquetes instalados. Al utilizar este comando, podrás activar uno de esos entornos y trabajar en él.
3. `jupyter notebook`: Este comando se utiliza para abrir el entorno de Jupyter Notebook. Jupyter Notebook es una aplicación web que te permite crear y compartir documentos interactivos que contienen código, visualizaciones y texto explicativo. Al ejecutar este comando, se abrirá una nueva pestaña del navegador con el entorno de Jupyter Notebook.
4. `spyder`: Al ejecutar este comando, se abrirá el entorno de desarrollo integrado (IDE) de Spyder. Spyder es un entorno de desarrollo muy utilizado en Python que ofrece características como edición de código, depuración y ejecución de programas.
5. `python`: Al ejecutar este comando, se iniciará el intérprete de Python en la línea de comandos. Podrás ejecutar comandos y scripts de Python directamente en la terminal.
6. `anaconda-navigator`: Con este comando, se abrirá la interfaz gráfica de Anaconda Navigator. Anaconda Navigator es una aplicación que te permite administrar tus entornos virtuales, instalar paquetes y lanzar aplicaciones como Jupyter Notebook y Spyder de manera visual.
7. `conda info`: Este comando proporciona información detallada sobre la instalación de Conda, como la versión de Conda, la ubicación de los archivos y la configuración actual.
8. `conda update conda`: Utilizando este comando, puedes actualizar la versión de Conda en tu sistema. Conda es el administrador de paquetes y entornos virtuales de Anaconda, y mantenerlo actualizado te permitirá acceder a las últimas funciones y mejoras.
9. `rm -rf ~/anaconda3`: Este comando se utiliza para desinstalar completamente Anaconda de tu sistema. Ten en cuenta que este comando eliminará permanentemente Anaconda y todos los archivos relacionados, incluyendo tus entornos virtuales y paquetes instalados. Asegúrate de hacer una copia de seguridad de tus datos importantes antes de ejecutar este comando.

Si tienes alguna pregunta o necesitas más ayuda, no dudes en dejar un comentario.

**¡Hasta la próxima!**

### Publicaciones Similares

Si te interesó este artículo, te recomendamos que explores otros blogs y recursos relacionados que pueden ampliar tus conocimientos. Aquí te dejo algunas sugerencias:

1. [🔗 Instalacion De Anaconda](#)
2. [🔗 Configurar Entorno Virtual Python Anaconda](#)
3. [🔗 01 Introducion A La Programacion Con Python](#)
4. [🔗 02 Variables Expresiones Y Statements Con Python](#)
5. [🔗 03 Objetos De Python](#)
6. [🔗 04 Ejecucion Condicional Con Python](#)
7. [🔗 05 Iteraciones Con Python](#)
8. [🔗 06 Funciones Con Python](#)
9. [🔗 07 Dataframes Con Python](#)
10. [🔗 08 Prediccion Y Metrica De Performance Con Python](#)
11. [🔗 09 Metodos De Machine Learning Para Clasificacion Con Python](#)
12. [🔗 10 Metodos De Machine Learning Para Regresion Con Python](#)
13. [🔗 11 Validacion Cruzada Y Composicion Del Modelo Con Python](#)

Esperamos que encuentres estas publicaciones igualmente interesantes y útiles. ¡Disfruta de la lectura!