

# Workflow Híbrido: VS Code Insiders + Google Colab

## Descripción General

Este tutorial te guía en el uso de un **workflow híbrido** que combina lo mejor de dos mundos:

- **VS Code Insiders:** Editor local potente para editar archivos Jupyter Notebook (.ipynb)
- **Google Colab:** Infraestructura en la nube con GPU/TPU gratuitas para ejecutar código

## Objetivo

Editar notebooks localmente en tu máquina Manjaro Linux, ejecutarlos remotamente en Google Colab aprovechando sus recursos de cómputo, y sincronizar automáticamente los resultados.

---

## Requisitos Previos

### Software Instalado

- **VS Code Insiders** - Editor de código
- **rlone** - Herramienta de sincronización con Google Drive
- **Cuenta de Google** - Para acceder a Google Drive y Colab

### Configuración Completada

- rclone configurado con Google Drive (remoto: **gdrive**)
- Scripts de sincronización creados en el directorio del proyecto

---

## Estructura del Proyecto

```
Defensa-Proyecto/  
  Telco-Customer-Churn.ipynb      # Notebook principal  
  WA_Fn-UseC_-Telco-Customer-Churn.csv # Dataset  
  sync-to-colab.sh                # Script: Local → Drive  
  sync-from-colab.sh              # Script: Drive → Local  
  sync-bidirectional.sh           # Script: Sincronización bidireccional  
  workflow-hibrido-colab.md       # Este tutorial
```

---

## Workflow Completo

### Paso 1: Editar Localmente en VS Code Insiders

1. Abrir VS Code Insiders

```
code-insiders /home/bootcamp/Proyectos-2026/Proyectos-Varios/BootCampVirtIA/Semana-05/Defensa-Proyecto
```

## 2. Abrir el notebook

- Navega a `Telco-Customer-Churn.ipynb`
- Haz clic para abrirlo

## 3. Editar el contenido

- Modifica celdas de código
- Agrega celdas de Markdown
- Reorganiza el notebook según necesites

## 4. Guardar cambios

- Presiona `Ctrl + S`
  - O usa: Archivo → Guardar
- 

## Paso 2: Sincronizar a Google Drive

### 1. Abrir terminal en el directorio del proyecto:

```
cd /home/bootcamp/Proyectos-2026/Proyectos-Varios/BootCampVirtIA/Semana-05/Defensa-Proyecto
```

### 2. Ejecutar script de sincronización:

```
./sync-to-colab.sh
```

### 3. Confirmar la sincronización:

- El script te preguntará: `¿Deseas continuar con la sincronización? (s/N):`
- Escribe `s` y presiona Enter

### 4. Esperar la subida:

- Verás el progreso de la sincronización
  - Al finalizar verás: `Sincronización completada exitosamente`
- 

## Paso 3: Ejecutar en Google Colab

### 1. Abrir Google Colab

- Ve a: <https://colab.research.google.com/>
- Inicia sesión con tu cuenta de Google

### 2. Abrir el notebook desde Drive

- Haz clic en Archivo → Abrir notebook
- Selecciona la pestaña Google Drive
- Navega a: Colab Notebooks/Defensa-Proyecto/

- Haz clic en: `Telco-Customer-Churn.ipynb`
3. **Configurar el entorno de ejecución** (opcional)
    - Ve a: Entorno de ejecución → Cambiar tipo de entorno de ejecución
    - Selecciona:
      - **Tipo de entorno:** Python 3
      - **Acelerador por hardware:** GPU (o TPU si lo necesitas)
    - Haz clic en Guardar
  4. **Ejecutar el notebook**
    - **Ejecutar todas las celdas:** Entorno de ejecución → Ejecutar todas
    - **Ejecutar celda por celda:** Shift + Enter en cada celda
    - **Ejecutar hasta cierto punto:** Selecciona celdas y usa Ctrl + Enter
  5. **Monitorear la ejecución**
    - Observa los outputs de cada celda
    - Revisa gráficos y resultados
    - Verifica que no haya errores
  6. **Guardar el notebook con resultados**
    - Presiona Ctrl + S
    - O usa: Archivo → Guardar
    - Los outputs quedan guardados en el archivo `.ipynb`
- 

#### Paso 4: Descargar Resultados a Local

1. **Volver a la terminal local:**

```
cd /home/bootcamp/Proyectos-2026/Proyectos-Varios/BootCampVirtIA/Semana-05/Defensa-Proyecto
```

2. **Ejecutar script de descarga:**

```
./sync-from-colab.sh
```

3. **Confirmar la descarga:**

- El script te preguntará: ¿Deseas continuar con la descarga? (s/N):
- Escribe s y presiona Enter

4. **Esperar la descarga:**

- Verás el progreso de la sincronización
  - Al finalizar verás: Descarga completada exitosamente
-

## Paso 5: Ver Resultados en VS Code Insiders

1. **Abrir VS Code Insiders** (si no está abierto)
  2. **Abrir el notebook actualizado**
    - El archivo `Telco-Customer-Churn.ipynb` ahora contiene:
      - Outputs de las celdas ejecutadas
      - Gráficos generados
      - Resultados de los modelos
      - Tablas y DataFrames
  3. **Revisar los resultados**
    - Desplázate por el notebook
    - Analiza los gráficos
    - Revisa las métricas de los modelos
  4. **Continuar editando** (si es necesario)
    - Modifica celdas
    - Agrega análisis adicional
    - Repite el ciclo desde el Paso 2
- 

## Atajos Útiles en VS Code Insiders

### Navegación

- **Ctrl + P**: Búsqueda rápida de archivos
- **Ctrl + Shift + E**: Explorador de archivos
- **Ctrl + B**: Mostrar/ocultar barra lateral

### Edición de Notebooks

- **Esc**: Modo comando
  - **Enter**: Modo edición
  - **A**: Insertar celda arriba (modo comando)
  - **B**: Insertar celda abajo (modo comando)
  - **M**: Cambiar a celda Markdown (modo comando)
  - **Y**: Cambiar a celda de código (modo comando)
  - **DD**: Eliminar celda (modo comando)
  - **Ctrl + S**: Guardar
- 

## Scripts de Sincronización

1. `sync-to-colab.sh` - Subir a Drive

Uso:

```
./sync-to-colab.sh
```

#### Qué hace:

- Sube archivos modificados desde tu máquina local a Google Drive
- Excluye archivos innecesarios (.git, .venv, **pycache**, etc.)
- Muestra progreso en tiempo real
- Confirma cuando la sincronización está completa

#### Cuándo usarlo:

- Después de editar el notebook localmente
  - Antes de ejecutar en Google Colab
  - Cuando quieras hacer backup de tu trabajo
- 

## 2. sync-from-colab.sh - Descargar desde Drive

#### Uso:

```
./sync-from-colab.sh
```

#### Qué hace:

- Descarga archivos desde Google Drive a tu máquina local
- Sobrescribe archivos locales con las versiones de Drive
- Excluye archivos innecesarios
- Muestra progreso en tiempo real

#### Cuándo usarlo:

- Después de ejecutar el notebook en Google Colab
- Para obtener resultados y outputs generados en Colab
- Para sincronizar cambios hechos en Drive

#### Advertencia:

- Este script sobrescribirá archivos locales si hay conflictos
  - Asegúrate de haber guardado cambios importantes antes de ejecutarlo
- 

## 3. sync-bidirectional.sh - Sincronización Bidireccional

#### Uso:

```
./sync-bidirectional.sh
```

#### Qué hace:

- Sincroniza archivos en ambas direcciones (Local Drive)
- Mantiene siempre la versión más reciente de cada archivo
- Detecta conflictos automáticamente

#### Primera vez:

Si es la primera vez que usas este script, necesitas inicializarlo:

```
rclone bisync "/home/bootcamp/Proyectos-2026/Proyectos-Varios/BootCampVirtIA/Semana-05/Defensa-Proyecto"
```

#### Cuándo usarlo:

- Cuando trabajas alternadamente entre local y Colab
- Para mantener sincronizados ambos entornos automáticamente
- Cuando no estás seguro de qué versión es más reciente

---

## Comandos Útiles de rclone

### Listar archivos en Google Drive

```
rclone ls gdrive:Colab\ Notebooks/Defensa-Proyecto
```

### Ver estructura de carpetas

```
rclone tree gdrive:Colab\ Notebooks/Defensa-Proyecto
```

### Copiar un archivo específico

```
rclone copy archivo.ipynb gdrive:Colab\ Notebooks/Defensa-Proyecto/
```

### Verificar diferencias

```
rclone check /ruta/local gdrive:Colab\ Notebooks/Defensa-Proyecto
```

### Ver espacio usado en Drive

```
rclone about gdrive:
```

---

## Solución de Problemas

Problema: “No se puede conectar a Google Drive”

Solución:

```
# Reconfigurar rclone  
rclone config reconnect gdrive:
```

---

Problema: “Archivos no se sincronizan”

Solución:

```
# Verificar configuración  
rclone config show  
  
# Probar conexión  
rclone lsd gdrive:  
  
# Sincronizar con verbose para ver detalles  
rclone sync /ruta/local gdrive:destino --verbose
```

---

Problema: “Conflictos de versiones”

Solución:

```
# Opción 1: Forzar subida (local sobrescribe Drive)  
rclone sync /ruta/local gdrive:destino --verbose  
  
# Opción 2: Forzar descarga (Drive sobrescribe local)  
rclone sync gdrive:destino /ruta/local --verbose  
  
# Opción 3: Hacer backup antes de sincronizar  
cp -r /ruta/local /ruta/backup  
./sync-from-colab.sh
```

---

**Problema:** “Script no tiene permisos de ejecución”

**Solución:**

```
chmod +x sync-to-colab.sh sync-from-colab.sh sync-bidirectional.sh
```

---

## Consejos y Mejores Prácticas

### 1. Guarda frecuentemente

- En VS Code: **Ctrl + S** después de cada cambio importante
- En Colab: **Ctrl + S** o **Archivo → Guardar**

### 2. Sincroniza antes y después

- **ANTES** de abrir en Colab: `./sync-to-colab.sh`
- **DESPUÉS** de ejecutar en Colab: `./sync-from-colab.sh`

### 3. Usa control de versiones

- Considera usar Git para versionar tu código
- Los notebooks con outputs pueden ser grandes, usa `.gitignore`

### 4. Organiza tus archivos

- Mantén datasets en una carpeta separada
- Usa subcarpetas para outputs, modelos, etc.

### 5. Aprovecha los recursos de Colab

- Usa GPU/TPU para entrenamientos pesados
- Descarga modelos entrenados a local
- Guarda checkpoints importantes en Drive

### 6. Documenta tu trabajo

- Usa celdas Markdown para explicar tu código
  - Agrega comentarios en el código
  - Documenta resultados y conclusiones
-



## Flujo de Trabajo Recomendado

### Para Desarrollo Iterativo

1. Editar código en VS Code (local)  
↓
2. Guardar (Ctrl + S)  
↓
3. Sincronizar a Drive (./sync-to-colab.sh)  
↓
4. Ejecutar en Colab (con GPU)  
↓
5. Guardar en Colab (Ctrl + S)  
↓
6. Descargar resultados (./sync-from-colab.sh)  
↓
7. Revisar en VS Code  
↓
8. Repetir desde paso 1

### Para Presentaciones/Demos

1. Ejecutar todo en Colab  
↓
  2. Verificar que todos los outputs estén correctos  
↓
  3. Guardar en Colab  
↓
  4. Descargar a local (./sync-from-colab.sh)  
↓
  5. Abrir en VS Code para presentar  
↓
  6. Exportar a PDF/HTML si es necesario
- 

## Recursos Adicionales

### Documentación Oficial

- Google Colab
- rclone Documentation
- VS Code Jupyter Extension

### Tutoriales Útiles

- Guía de Google Colab
  - rclone con Google Drive
-

## Próximos Pasos: Migración a Paperspace Gradient

Si quieres una experiencia más integrada (editar Y ejecutar desde VS Code), considera migrar a **Paperspace Gradient**:

### Ventajas

- Edición y ejecución directa desde VS Code
- GPU gratuitas (limitadas)
- Sin sincronización manual
- Terminal remoto integrado
- Experiencia nativa de desarrollo

### Configuración Rápida

```
# 1. Instalar extensión Remote-SSH
code-insiders --install-extension ms-vscode-remote.remote-ssh

# 2. Crear cuenta gratuita
# https://www.paperspace.com/gradient

# 3. Conectar desde VS Code
# Ctrl + Shift + P + "Remote-SSH: Connect to Host"
```

---

## Notas Finales

Este workflow híbrido te permite:

- **Editar** con la comodidad de VS Code
- **Ejecutar** con el poder de Google Colab
- **Sincronizar** automáticamente con rclone
- **Guardar** todo en Google Drive

¡Disfruta de lo mejor de ambos mundos!

---

**Autor:** Configuración automática

**Fecha:** 2025-11-19

**Versión:** 1.0