Slope Analysis Workflow Documentation

1. Software Requirements

- ArcGIS Pro (with Spatial Analyst license)
- ArcGIS Pro Python environment (ArcPy-enabled)

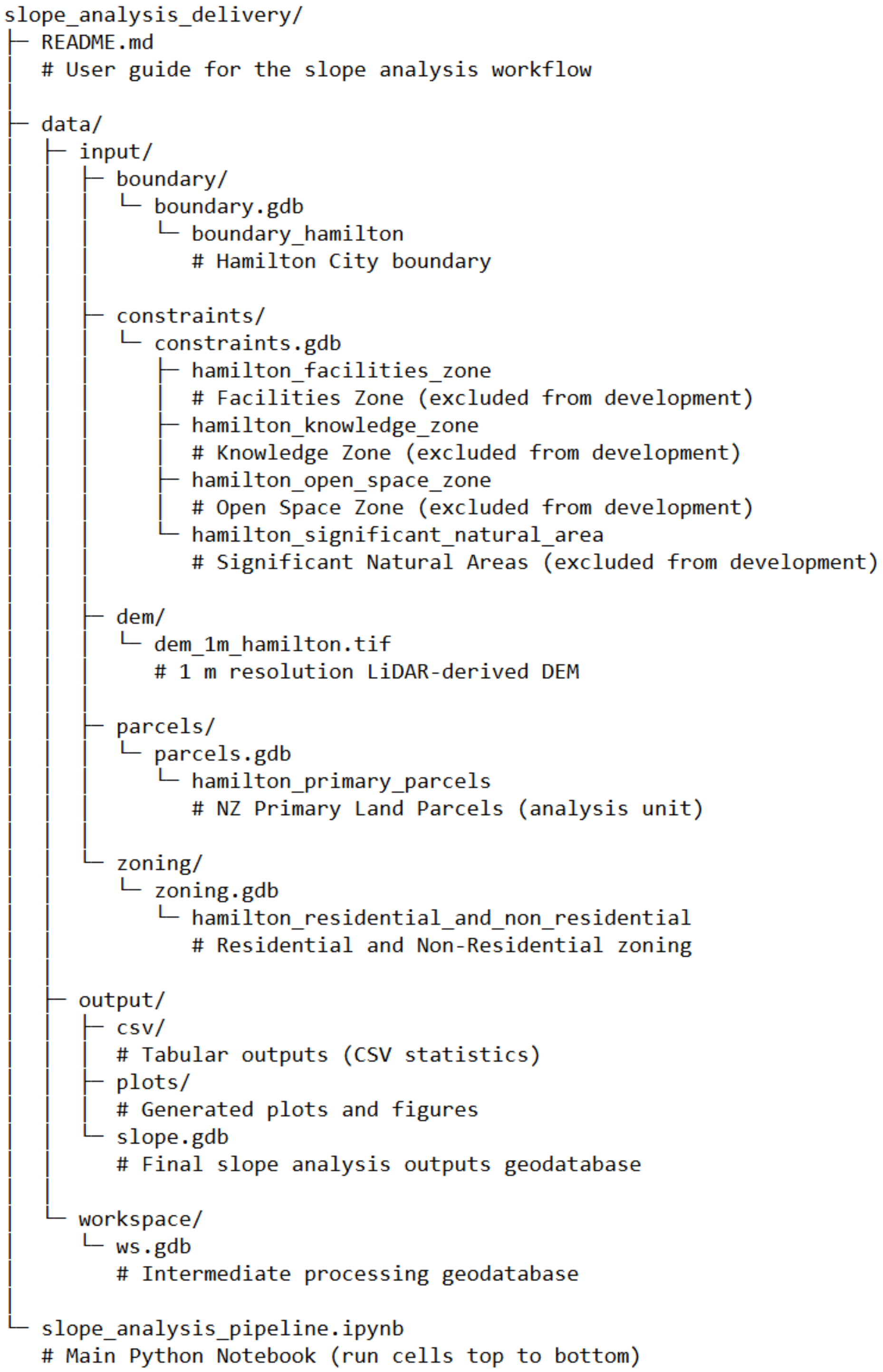
2. Documentation Structure

This section documents the project folder structure and provides a clear explanation of every dataset produced by the workflow, including its role in the analysis pipeline.

Figure 1 illustrates the overall project directory structure of the slope analysis workflow, showing the organisation of input data, intermediate processing outputs, and final delivery products.

The documentation is organised by data location:

- **data/input** – read-only source datasets
- **data/workspace** – intermediate, regenerable processing outputs
- **data/output** – final delivery datasets (spatial, tabular, and figures)



```
slope_analysis_delivery/
├─ README.md
│  # User guide for the slope analysis workflow
│
├─ data/
│  ├─ input/
│  │  ├─ boundary/
│  │  │  └─ boundary.gdb
│  │  │     └─ boundary_hamilton
│  │  │        # Hamilton City boundary
│  │  │
│  │  ├─ constraints/
│  │  │  └─ constraints.gdb
│  │  │     ├─ hamilton_facilities_zone
│  │  │     │  # Facilities Zone (excluded from development)
│  │  │     ├─ hamilton_knowledge_zone
│  │  │     │  # Knowledge Zone (excluded from development)
│  │  │     ├─ hamilton_open_space_zone
│  │  │     │  # Open Space Zone (excluded from development)
│  │  │     └─ hamilton_significant_natural_area
│  │  │        # Significant Natural Areas (excluded from development)
│  │  │
│  │  ├─ dem/
│  │  │  └─ dem_1m_hamilton.tif
│  │  │     # 1 m resolution LiDAR-derived DEM
│  │  │
│  │  ├─ parcels/
│  │  │  └─ parcels.gdb
│  │  │     └─ hamilton_primary_parcels
│  │  │        # NZ Primary Land Parcels (analysis unit)
│  │  │
│  │  └─ zoning/
│  │     └─ zoning.gdb
│  │        └─ hamilton_residential_and_non_residential
│  │           # Residential and Non-Residential zoning
│  │
│  ├─ output/
│  │  ├─ csv/
│  │  │  # Tabular outputs (CSV statistics)
│  │  ├─ plots/
│  │  │  # Generated plots and figures
│  │  └─ slope.gdb
│  │     # Final slope analysis outputs geodatabase
│  │
│  └─ workspace/
│     └─ ws.gdb
│        # Intermediate processing geodatabase
│
└─ slope_analysis_pipeline.ipynb
   # Main Python Notebook (run cells top to bottom)
```

Figure 1. Project directory structure of the slope analysis workflow.

2.1 data/input — Input datasets (read-only)

Location: data/input/

These datasets are **never modified** by the workflow and form the fixed analytical inputs.

Table 1 summarises all read-only input datasets used in the slope analysis workflow, including spatial boundaries, constraint layers, elevation data, parcel geometry, and zoning information.

Table 1. Read-only input datasets used in the slope analysis workflow

Subdirectory	Dataset	Type	Description
boundary/	boundary.gdb / boundary_hamilton	Feature Class (Polygon)	Hamilton City administrative boundary. Used to clip all spatial datasets to a consistent analysis extent.
constraints/	constraints.gdb	Geodatabase	Geodatabase containing constraint feature classes listed below.
	constraints.gdb / hamilton_facilities_zone	Feature Class (Polygon)	Facilities Zone constraint layer.
	constraints.gdb / hamilton_knowledge_zone	Feature Class (Polygon)	Knowledge Zone constraint layer.
	constraints.gdb / hamilton_open_space_zone	Feature Class (Polygon)	Open Space Zone constraint layer.
	constraints.gdb / hamilton_significant_natural_area	Feature Class (Polygon)	Significant Natural Area (SNA) constraint layer.
dem/	dem_1m_hamilton.tif	Raster	1 m resolution LiDAR-derived DEM. Source raster for slope calculation.
parcels/	parcels.gdb / hamilton_primary_parcels	Feature Class (Polygon)	NZ Primary Land Parcels. Core analysis unit for all parcel-level statistics.
zoning/	zoning.gdb / hamilton_residential_and_non_residential	Feature Class (Polygon)	Residential and Non-Residential zoning layer. Used to classify parcels via spatial join.

2.2 data/workspace — Intermediate processing outputs

Location: data/workspace/ws.gdb

This geodatabase stores **intermediate datasets** generated during processing.

Table 2 lists all intermediate datasets generated during the processing stage, together with their generating functions and roles within the analysis pipeline.

Table 2. Intermediate datasets generated in the processing workspace

Dataset	Type	Generated by	Description
constraints_merged	Feature Class (Polygon)	preprocess_parcels_by_constraints()	Merged constraint layer combining Significant Natural Areas, Open Space Zone, Knowledge Zone, and Facilities Zone. Provides a single constraint feature for subsequent dissolve and erase operations.
constraints_dissolved	Feature Class (Polygon)	preprocess_parcels_by_constraints()	Dissolved version of constraints_merged, removing internal boundaries to improve erase behaviour and produce cleaner geometry.
parcels_zoned	Feature Class (Polygon)	run_spatial_join_zone_to_parcels()	Parcel polygons with zoning attributes (Residential / Non-Residential) attached using Spatial Join. Acts as the zoned parcel base for downstream analysis.
parcels_erased_constraints	Feature Class (Polygon)	preprocess_parcels_by_constraints()	Parcel polygons with constraint areas removed (parcels_zoned minus constraints_dissolved), representing potentially developable land area.
slope_degree	Raster (Continuous)	run_slope_and_zonal()	Continuous slope raster (degrees) derived from the DEM using arcpy.sa.Slope. Serves as the base raster for zonal statistics and slope classification.
zonal_all	Table	run_slope_and_zonal()	Output table from ZonalStatisticsAsTable(..., statistics_type="ALL"). Only MEAN and MAX are joined

			back to parcel datasets; the table itself is not used in downstream analysis.
parcel_slope_class_area	Table	calc_slope_class_area_by_parcel()	Output of TabulateArea, storing parcel × slope-class area statistics (m²) for each defined slope bin.
slope_class_raster_all_bins	Raster (Classified)	calc_slope_class_area_by_parcel()	Reclassified slope raster for all slope bins (0–6, 6–12, …, 35–90 degrees). Saved for QA and cartographic visualisation only; not required for downstream analysis.
city_level_steep_slope_classified_raster	Raster (Classified)	export_steep_slope_zones()	Reclassified steep slope raster (≥18°) with class codes: 1 = 18–25, 2 = 25–35, 3 = 35–90. Saved for QA / validation only.

2.3 data/output — Final delivery outputs

Location: data/output/

These datasets are **final deliverables** intended for inspection, reporting, and downstream analysis.

2.3.1 Spatial outputs (Feature Classes)

Location: data/output/slope.gdb

Table 3 describes all final spatial feature classes delivered by the workflow. These datasets are intended for inspection, mapping, and downstream spatial analysis.

Table 3. Final spatial output feature classes produced by the slope analysis workflow

Dataset	Type	Generated by	Description
parcels_slope	Feature Class (Polygon)	run_slope_and_zonal() + calc_slope_class_area_by_parcel()	Full parcel-level output dataset containing original parcel attributes, zonal slope statistics (MEAN, MAX), and slope-class area fields (AREA_SLOPE_*, m²).
parcels_slope_deliver	Feature Class (Polygon)	build_parcels_slope_delivery_fc()	Cleaned delivery dataset produced using a whitelist field strategy; designed as a stable schema for CSV export and pandas-based analysis.
city_level_steep_slope_zones	Feature Class (Polygon)	export_steep_slope_zones()	Contiguous steep slope zones (≥18°) classified into steepness bins and annotated with slope_class labels.
parcel_level_steep_slope_zones	Feature Class (Polygon)	export_steep_slope_zones()	Steep slope polygons intersected with parcel boundaries; each feature represents a portion of a parcel occupied by a specific steep slope class.
parcels_affected_by_steep_slopes	Feature Class (Polygon)	export_steep_parcels()	Whole parcels flagged as affected if any steep slope area (≥18°) exists within the parcel.

2.3.2 Tabular outputs (CSV)

Location: data/output/csv/

Table 4 summarises all analysis-ready CSV outputs generated from parcel-level and aggregated statistics. These tables are used directly by the post-processing statistics scripts and plotting scripts.

Table 4. Analysis-ready tabular outputs (CSV) generated for reporting and plotting

File	Description
parcels_slope_steep_metrics.csv	Main parcel-level delivery CSV with slope and steep metrics
city_slope_stats.csv	City-wide slope composition (area and percentage)
zone_slope_stats.csv	Zone-level slope and steep exposure metrics
zone_slope_stats_res_vs_nonres.csv	Residential vs Non-Residential comparison
parcel_steep_stats.csv	Parcels ranked by steep exposure
parcel_steep_bin_counts_0_100_res_nonres.csv	STEEP_BIN distribution (all / res / non-res)

2.3.3 Figures (PNG)

Location: data/output/plots/

Table 5 lists all publication-ready figures generated by the plotting script. All figures are derived exclusively from CSV outputs and saved at 300 dpi for reporting and presentation.

Table 5. Publication-ready figures generated from slope analysis outputs

Figure	Description
fig_city_slope_distribution.png	City-wide slope composition
fig_residential_slope_pie.png	Residential zone slope composition
fig_nonresidential_slope_pie.png	Non-Residential zone slope composition
fig_steep_total_comparison.png	Total steep exposure (≥18°) by zone
fig_steep_composition_comparison.png	Steep-bin composition by zone
fig_steep_bin_pie_all_parcels.png	Parcel STEEP_BIN distribution (all)
fig_steep_bin_pie_residential.png	Parcel STEEP_BIN distribution (residential)
fig_steep_bin_pie_non_residential.png	Parcel STEEP_BIN distribution (non-residential)
fig_steep_bin_counts_res_vs_nonres.png	Res vs Non-Res parcel counts by STEEP_BIN

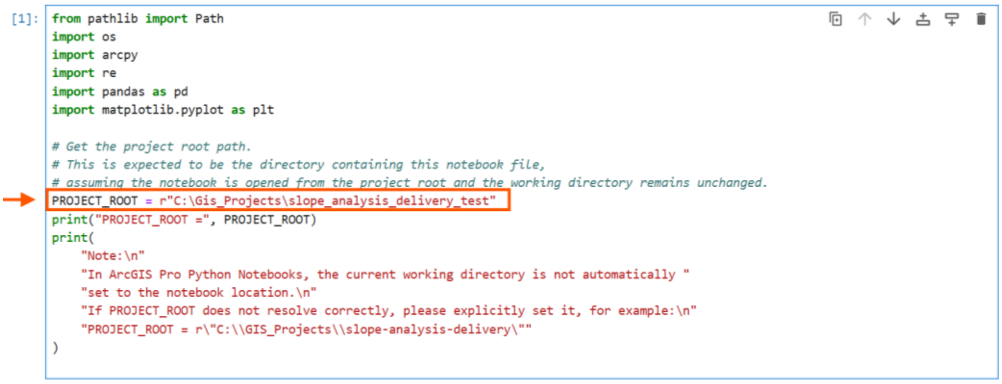
2.4 Note on data deletion and regeneration:

- `data/input/` contains all required input datasets and should not be deleted.
- `data/workspace/` contains intermediate processing outputs and is fully regenerable.
- `data/output/` contains final delivery datasets, but these are also reproducible.

Re-running the notebook `slope_analysis_pipeline.ipynb` from top to bottom will fully regenerate all intermediate and output datasets, as long as the input data under `data/input/` remain unchanged.

3. How to run

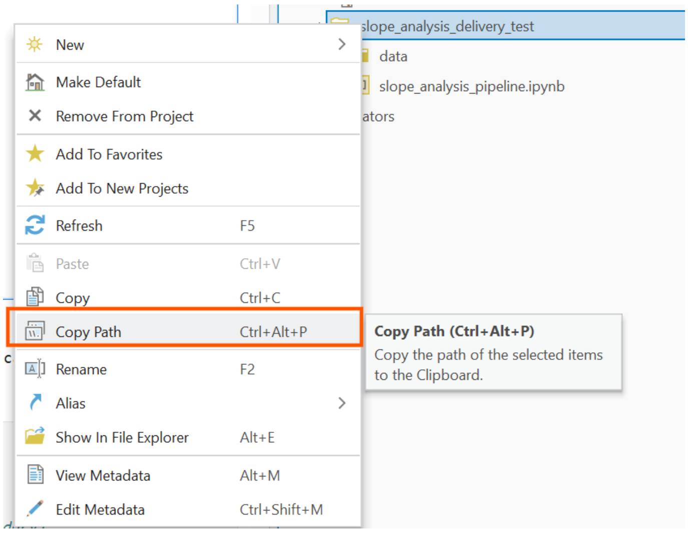
1) Open the Python Notebook `slope_analysis_pipeline.ipynb` in ArcGIS Pro.
2) Modify the PROJECT_ROOT value
   a) In the first cell of the notebook, set the value of `PROJECT_ROOT` to the absolute path of the `slope_analysis_delivery` directory.



```
[1]: from pathlib import Path
     import os
     import arcpy
     import re
     import pandas as pd
     import matplotlib.pyplot as plt

     # Get the project root path.
     # This is expected to be the directory containing this notebook file,
     # assuming the notebook is opened from the project root and the working directory remains unchanged.
     PROJECT_ROOT = r"C:\Gis_Projects\slope_analysis_delivery_test"
     print("PROJECT_ROOT =", PROJECT_ROOT)
     print(
         "Note:\n"
         "In ArcGIS Pro Python Notebooks, the current working directory is not automatically "
         "set to the notebook location.\n"
         "If PROJECT_ROOT does not resolve correctly, please explicitly set it, for example:\n"
         "PROJECT_ROOT = r\"C:\\GIS_Projects\\slope-analysis-delivery\""
     )
```

   b) You can obtain the path by right-clicking the `slope_analysis_delivery` folder in ArcGIS Pro and selecting **Copy Path**, then paste it inside the raw string (`r"..."`) assigned to `PROJECT_ROOT`.



Copy Path (Ctrl+Alt+P)
Copy the path of the selected items to the Clipboard.

3) Run all cells in the notebook from top to bottom. Execution time depends on data size and hardware performance. The full workflow may take several minutes to complete.