

# Automating CorrDiff Model Analysis



A Toolkit for Evaluating, Processing, and Visualizing Weather Model Predictions

An Explainer and Guide for the `corrdiff\_output` Repository

# A Comprehensive Toolkit for Post-Processing Experiment Results



## Mask Datasets

Apply landmasks to filter and refine weather model data for accurate analysis.



## Generate Visual Plots

Create a suite of publication-ready visualizations for comprehensive model evaluation.



## Compute Key Metrics

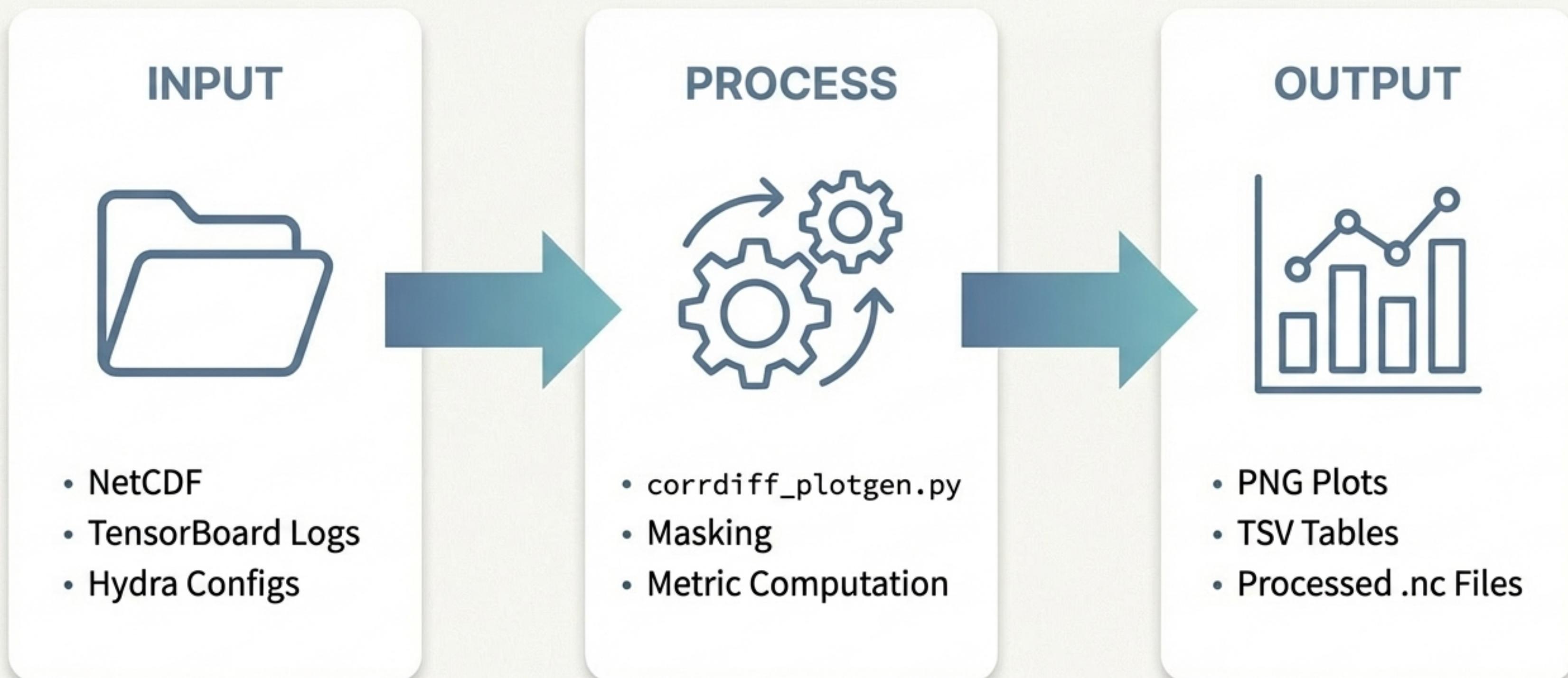
Automatically calculate standard evaluation metrics for both regression and diffusion models.

	=	=	=	=
	...	...	...	...
	..	...	...	....
	...	...	...	...
	...	...	...	...

## Export Processed Data

Organize and export computed metrics and processed data into structured tables.

# The Core Workflow: From Raw Data to Insight



# Getting Started: Installation and Dependencies

## Create Conda Environment



```
# Install the required dependencies  
from the provided file  
conda env create -f corrdiff.yml
```

## Additional Dependencies



```
# If not included in the environment  
file  
pip install xskillscore
```

# Generate All Outputs with a Single Command

PROCESS



Path to your data folder containing  
NetCDF, TensorBoard logs, and  
Hydra configs.

Path where all generated plots  
and tables will be saved.

```
python corrdiff_plotgen.py <input_directory>  
<output_directory> --n-ensemble=1 --masked=no
```

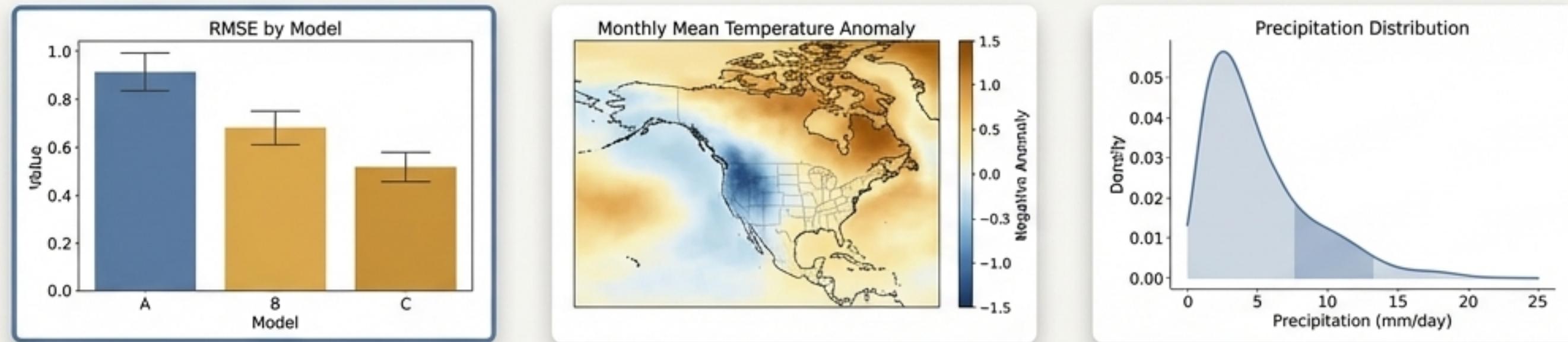
Specify the number of  
ensemble members (default  
is 1).

Apply landmasking to the data  
( 'yes' or 'no', default is 'yes').

# A Rich Suite of Generated Outputs



## Plots (.png)



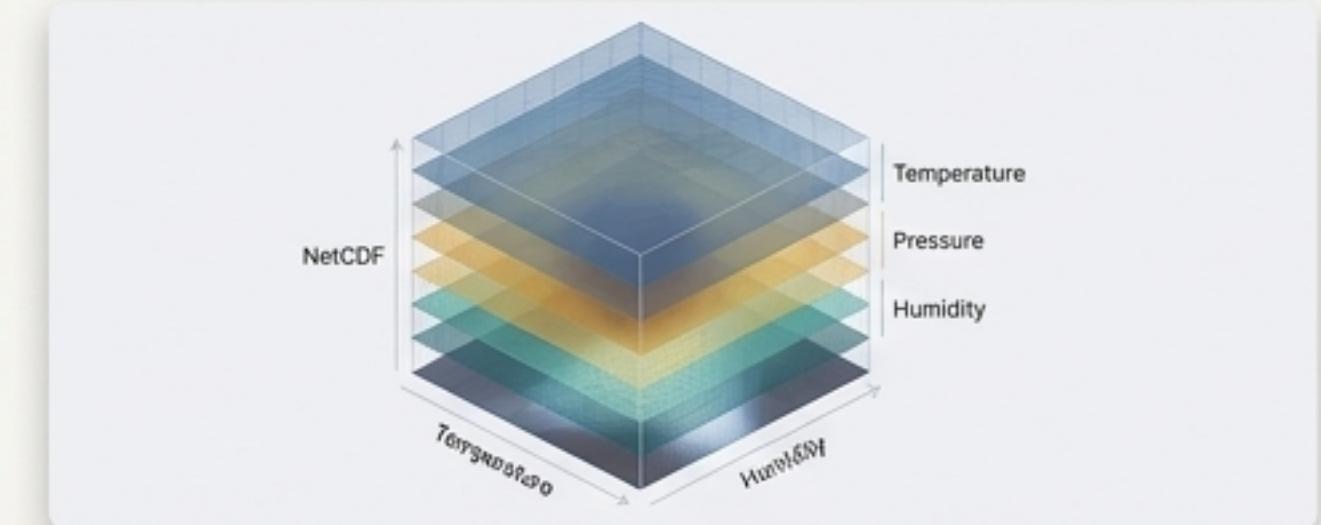
Metrics, monthly errors, and PDF distributions.

## Tables (.tsv)

	Jan	Feb	Mar
RNSE	1.621e+02	1.758e+01	2.583e+01
CRPS	1.648e+02	1.803e+02	2.327e+02
RNSE	1.693e+02	1.482e+02	1.773e+02
ORSE	1.789e+02	1.365e+02	1.783e+02
CRPS	1.785e+02	1.252e+01	1.181e+02
CRPSK	6.363e+02	5.883e+01	4.938e+01
KODEN	4.522e+02	5.853e+01	8.985e+01
RUREC	4.541e+02	3.591e+01	4.475e+03

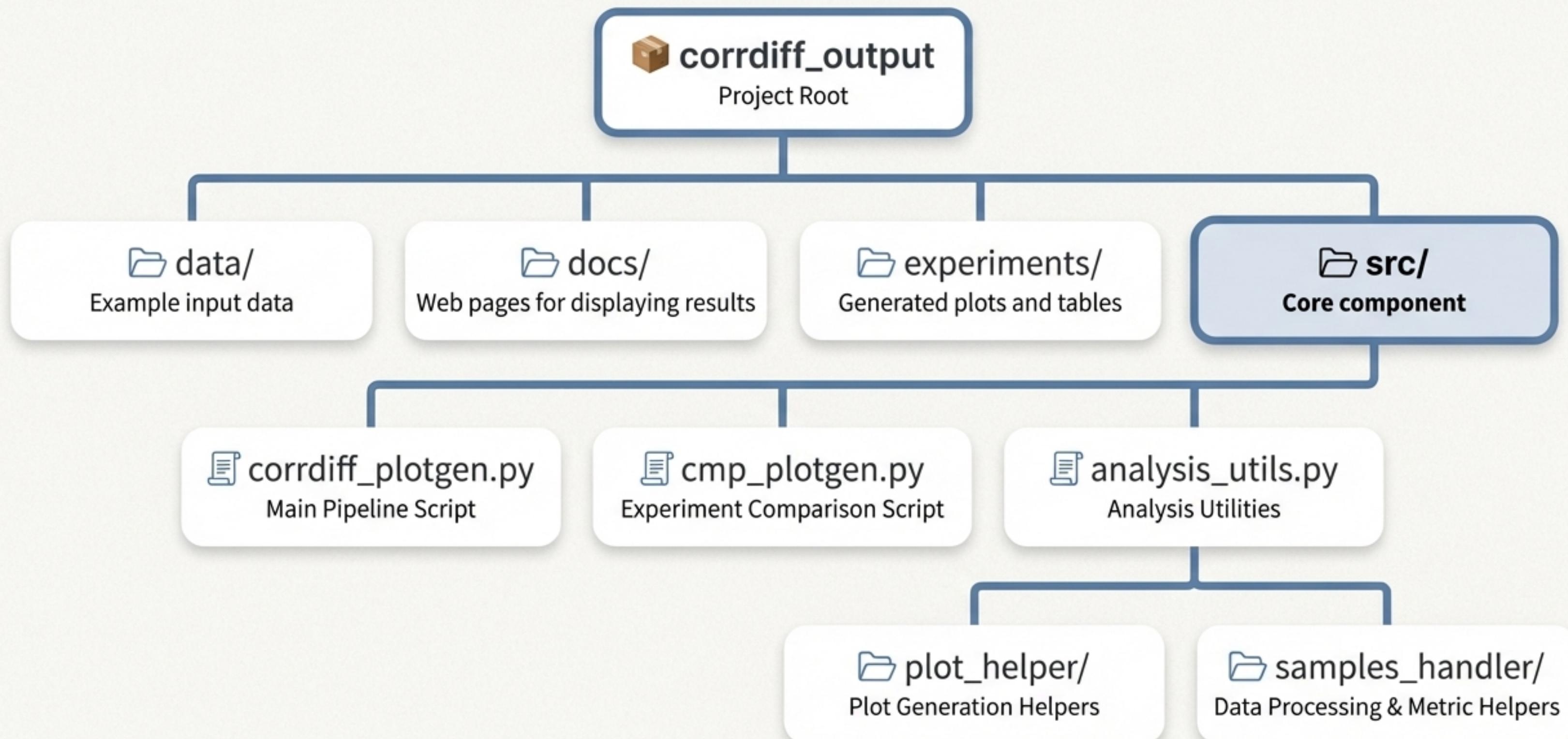
Mean metrics and monthly grouped metrics.

## Processed Data (.nc)



Masked and processed NetCDF datasets.

# Under the Hood: The Project Architecture



# The Main Pipeline: `corrdiff\_plotgen.py`



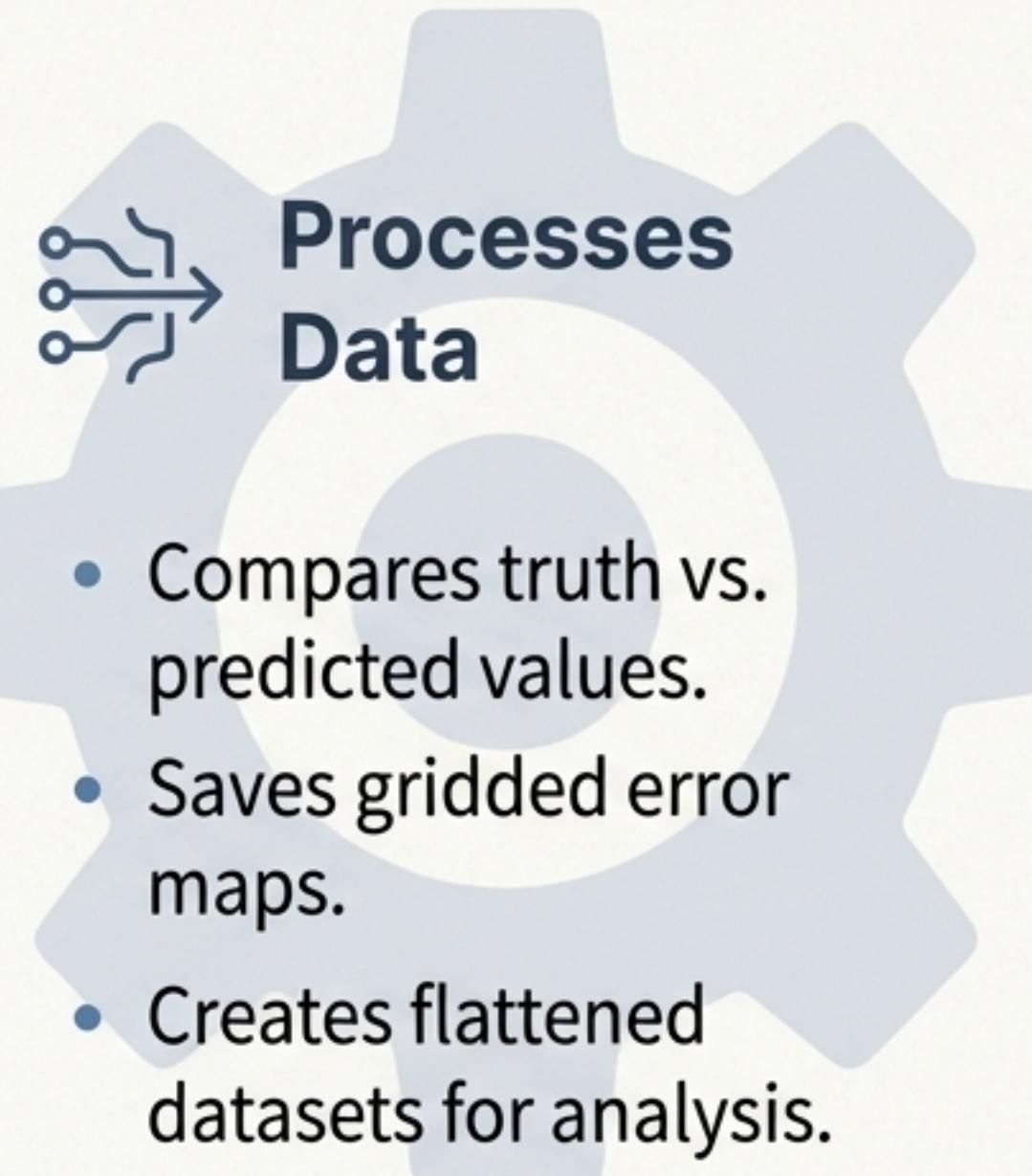
- </> **Reads:** Ingests NetCDF weather model data from the specified input directory.
- Computes:** Orchestrates the calculation of all model evaluation metrics.
- Generates:** Produces the final visual plots and structured data tables.

# The Computation Core: `samples\_handler`



## Computes Core Metrics

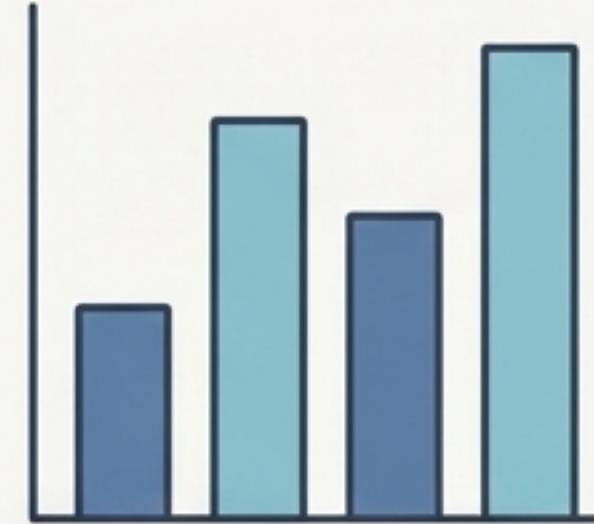
- Root Mean Square Error (RMSE)
- Continuous Ranked Probability Score (CRPS)
- Standard Deviation



## Identifies Extremes

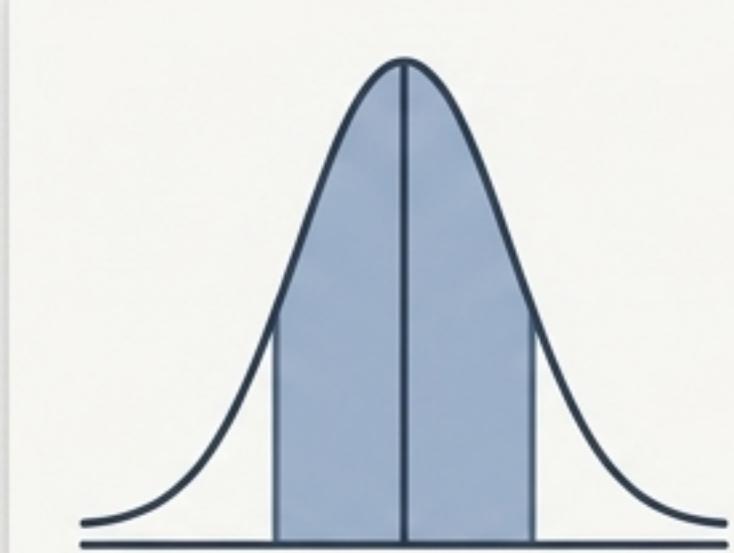
- Selects overall top dates based on error metrics.
- Creates p90 grids per N-year bin for extreme event analysis.

# The Visualization Suite: `plot\_helper`



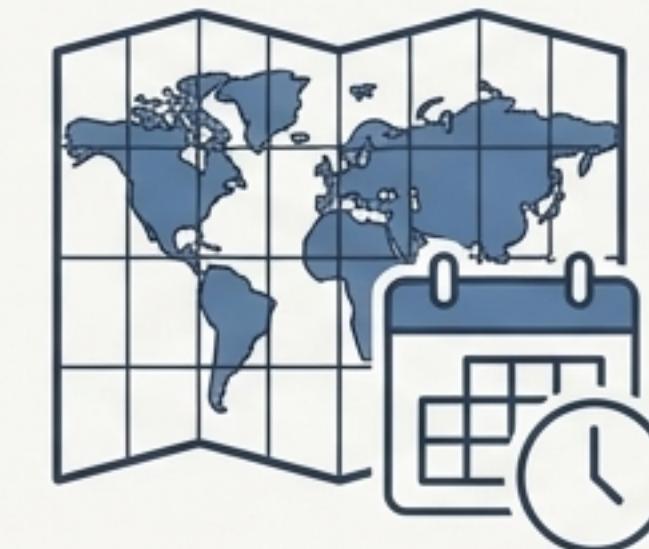
## Metric Summaries

Generates clear bar charts comparing key evaluation metrics.



## PDF Distributions

Plots probability density functions for critical weather variables.



## Spatio-Temporal Errors

Creates monthly and decadal error map visualizations to identify patterns.

# The Utility Belt: `analysis\_utils.py`



## Data Export

Provides wrappers for exporting metric tables to TSV format.



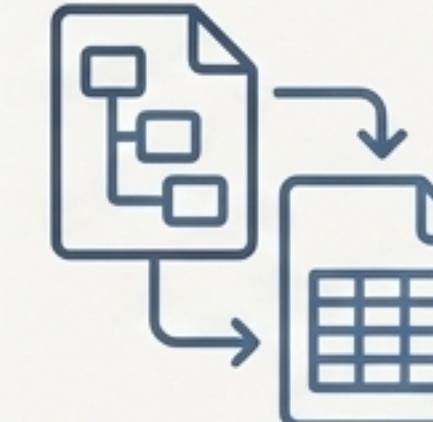
## Time Binning

Groups time-indexed metric datasets into fixed N-year bins for long-term analysis.



## Log Parsing

Includes functions for TensorBoard scalar extraction and plotting training loss.



## Config Handling

Flattens Hydra YAML configurations and converts them to TSV for easy comparison.

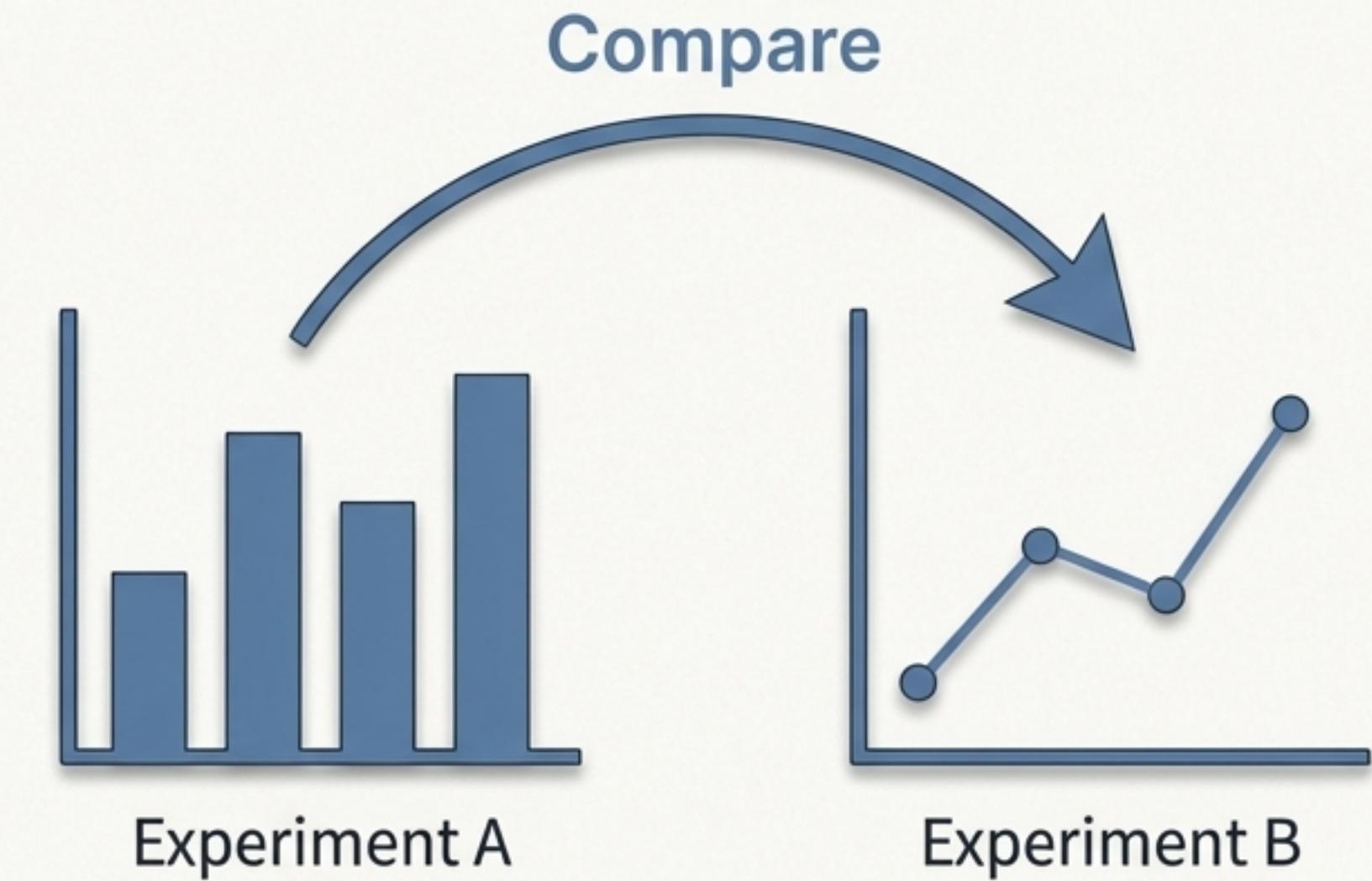
# Beyond a Single Run: Comparing Experiments with `cmp\_plotgen.py`

## Purpose

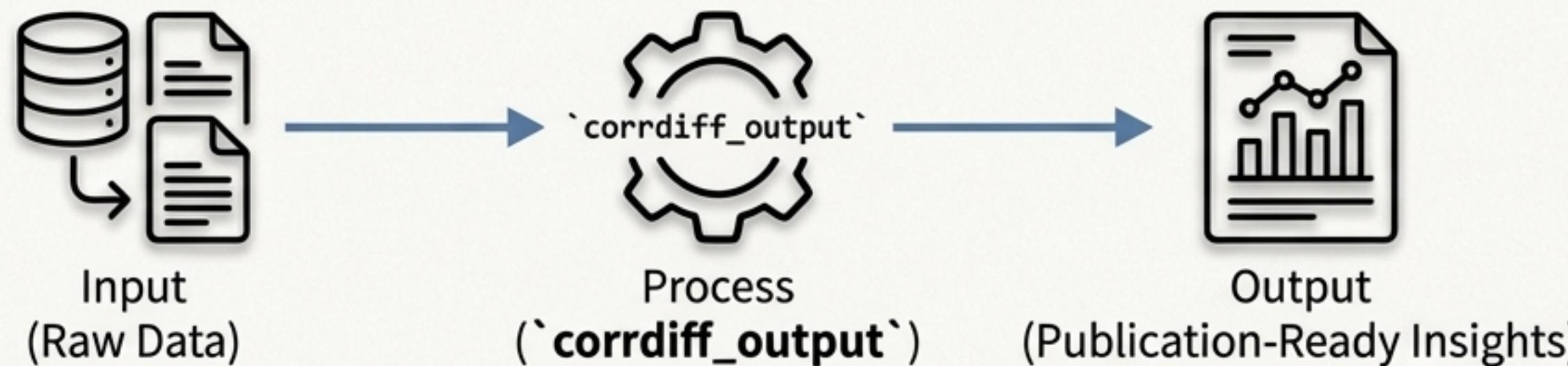
Designed specifically to compare metrics across multiple experiment output directories.

## Outputs

- **Bar Charts:** For direct, side-by-side metric comparisons among experiments.
- **Line Charts:** For visualizing decadal metric trends across different model runs.



# `'corrdiff_output'`: From Raw Data to Publication-Ready Insights



## Automate

Eliminate tedious and error-prone manual post-processing.

## Standardize

Apply a consistent set of metrics and visualizations across all your experiments.

## Accelerate

Go from raw model output to deep analytical insight with a single command.



Code  
Processing



Plots



Tables