How to dig up results from a big set of data?

Azin Farahani
HumanIC Project · Granlund Oy · 02 July 2026

BLOCK 1
Data &
Statistics

BREAK
5 min

BLOCK 2
Predictive
Models

Granlund

You have years of sensor data from thousands of buildings,

Where do you even start?

What would you want to know, and how would you approach it?

What Is Data? Types We Encounter in Buildings

The data type determines the analysis methods; you cannot apply the same approach to all of them

Time Series

Indoor temperature · Energy consumption · CO_2 · Occupancy sensors · Weather

Continuous, hourly or sub-hourly. Most common in building monitoring.

Tabular

Building characteristics · Energy certificates · Survey responses · Maintenance logs

One row per building or event. Good for cross-sectional analysis.

Images

Thermal camera scans · Facade inspection photos · Floor plan scans · Satellite imagery

Requires computer vision. Growing use in fault detection.

Point Clouds

Laser scanning for BIM · Geometry capture · As-built documentation

3D spatial data. Used in renovation and digital twin workflows.

Text / Logs

Fault reports · Occupant complaints · Maintenance records · BMS alarms

Data Cleaning & Preprocessing

Garbage in, garbage out, a sophisticated model on messy data gives wrong answers with high confidence

Missing values

Sensor dropout, communication failures, manual gaps.
pandas: df.isnull() · dropna() · fillna()
Decision: interpolate, flag, or remove, depends on gap length

Outliers

Faulty sensors, manual entry errors, physical impossibilities.
scipy: z-score · IQR method
Decision: remove, cap, or investigate, never ignore silently

Noise

Measurement uncertainty, sensor vibration, electrical interference.
numpy/pandas: rolling mean · Savitzky-Golay filter
Careful: smoothing can hide real peaks

Timestamp issues

Different sampling rates, timezone errors, daylight saving gaps.
pandas: resample() · tz_localize() · asfreq()
Most common hidden problem in merged datasets

Data Preprocessing: Aggregation, Interpolation & Data Generation

These choices are not neutral; they change what the data says. Document every step.

Aggregation

Hourly → daily → monthly → seasonal
Apartment-level → building-level → district-level
pandas: resample('D').mean() · groupby()
Key: hourly data catches peak overheating, daily averages hide it

Interpolation

Filling gaps in time series: linear, spline, forward-fill
pandas: interpolate(method='linear') · ffill()
Rule of thumb: interpolate short gaps (≤ 6h), flag long ones

Data generation for buildings

Synthetic data from EnergyPlus / IDA ICE, when measured data is scarce
Parametric simulation sweeps as ML training data
Weather morphing for future climate scenarios
→ Direct link to the simulation session

Data Visualisation: Explore Before You Analyse

Visualisation reveals patterns, outliers, and structures that statistics alone can miss

Time series plot

See trends, seasonality, anomalies, gaps

matplotlib · plotly

Indoor temperature over summer — when does it spike?

Heatmap

Patterns across two dimensions simultaneously

seaborn · plotly

Hour of day × day of week energy consumption matrix

Distribution / histogram

Shape of data — normal? skewed? bimodal?

seaborn · matplotlib

Distribution of degree hours across 6,000 apartments

Scatter plot / correlation

Relationship between two variables

seaborn: pairplot()

Outdoor temp vs indoor temp — how strong is the link?

Why Statistical Analysis? What Questions It Answers

Statistics describes, compares, and finds relationships, it is the foundation before any model

Descriptive

What does the data look like?
Distribution · median · percentiles · variance

What % of Helsinki apartments exceeded 27°C in summer 2021?

pandas: describe() · value_counts()

Comparative

Is A different from B?
Are new buildings better than old ones?

Do post-2012 code buildings have fewer degree hours than pre-2012?

scipy: ttest_ind() · mannwhitneyu()

Relational

Does X predict Y?
Correlation, regression, moving averages

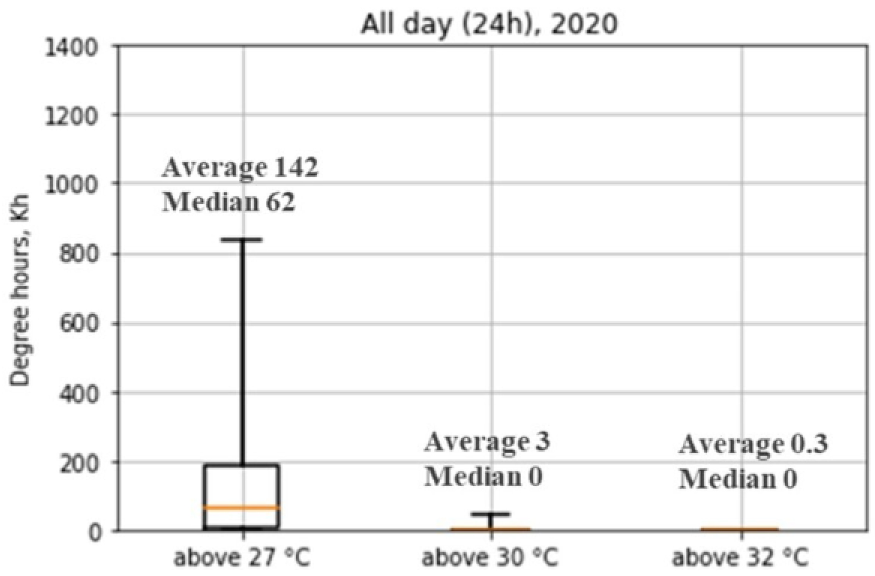
How strongly does outdoor 5-day moving average temperature predict indoor temperature?

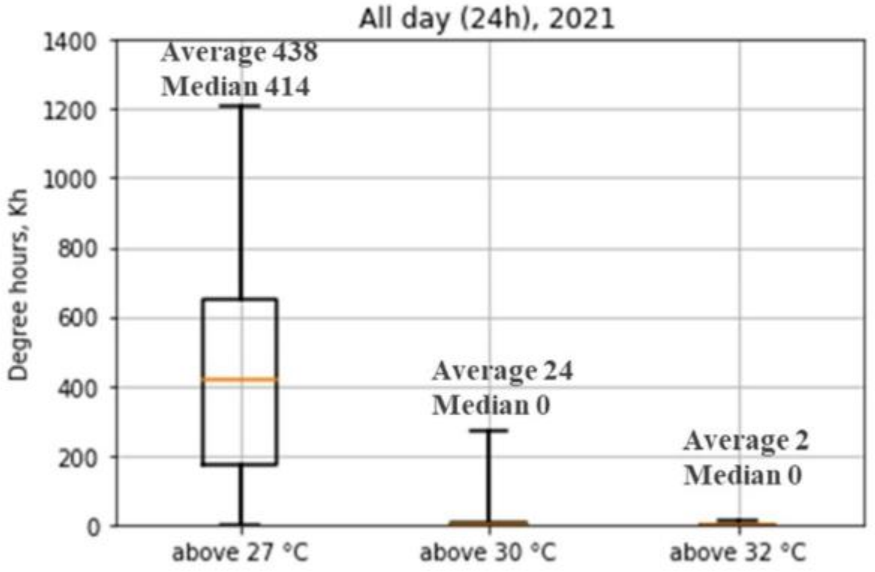
pandas: corr() · statsmodels OLS

Example: Overheating Assessment Across Building Stock

Dataset & method:

- Large field measurement dataset , multiple summers, Helsinki apartments
- Compared average summer vs hot summer (2020, 2021)
- Statistical characterisation: degree hours distribution across building stock
- Identified which buildings, floors, and orientations were most at risk
- Tools: pandas · scipy · seaborn





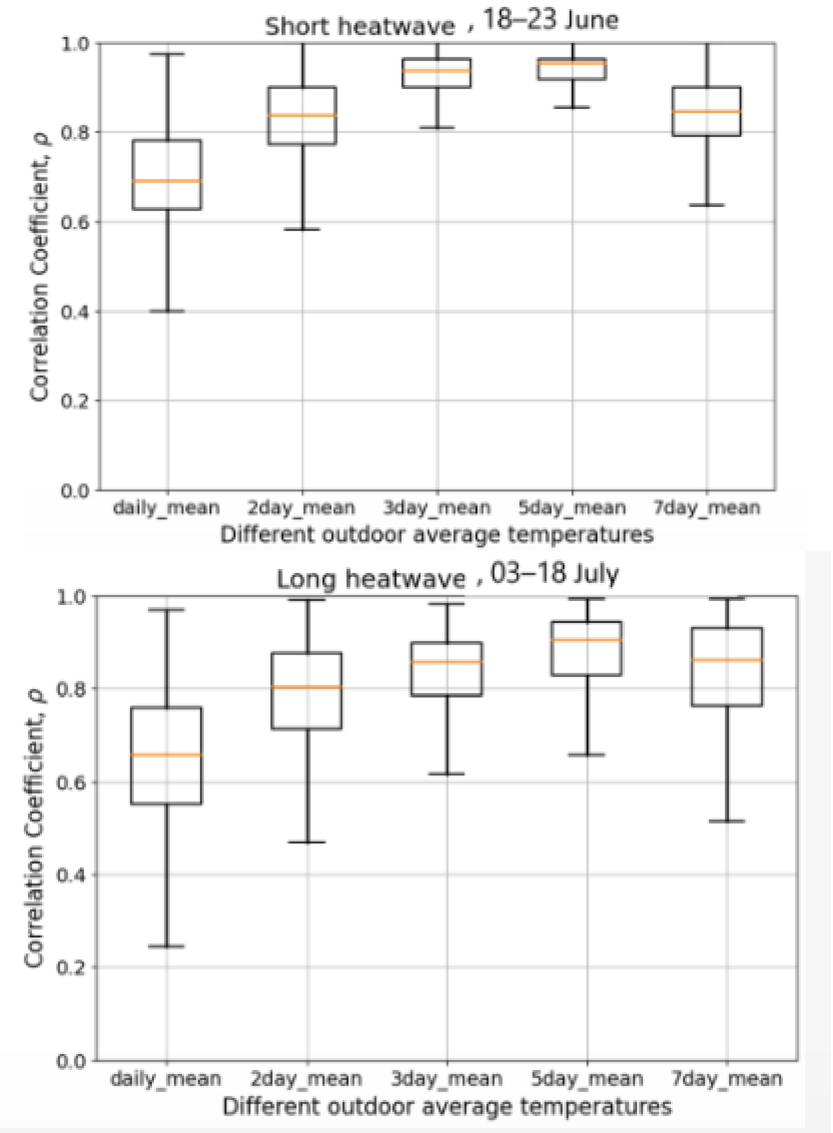
Building on Innovation

Granlund

Example: Outdoor–Indoor Temperature Correlation

Dataset & method:

- 6,000+ apartments in Helsinki, hot summer 2021 heatwaves
- Hourly indoor temperatures + outdoor weather data
- 5-day moving average outdoor temperature + 7-day moving average solar radiation
- Pearson correlation, linear regression
- Tools: pandas · scipy · statsmodels · seaborn



5 min Break

Statistics revealed what was happening and why,

What could statistics NOT tell us, and what would we need instead?

Predictive Methods: Why and Where in Buildings

All use cases share one capability, learning from historical data to predict new situations

Demand response

Predict when peak load will occur
Shift consumption to off-peak

Smart controls

Pre-heat/cool before occupancy
MPC, model predictive control

Design tools

Predict energy performance
before the building is built

Fault detection

Predict equipment failure
Flag anomalies automatically

Monitoring

Flag when building deviates
from expected behaviour

Optimisation

Continuously improve setpoints
from operational feedback

Supervised vs Unsupervised Learning

SUPERVISED

You have inputs AND known outputs
Model learns the mapping

Regression
Predict indoor temperature, energy demand
Linear regression · Random forest · XGBoost · Neural networks

Classification
Is this building overheating? yes/no
Decision tree · SVM · Gradient boosting

Needs labelled data, costly to obtain
but gives directly actionable predictions

UNSUPERVISED

Only inputs , model finds structure
No labels needed

Clustering
Group buildings by behaviour pattern
K-means · DBSCAN · Hierarchical clustering

Anomaly detection
Flag unusual energy consumption without labelled faults
Isolation forest · Autoencoders

Finds patterns you didn't know to look for
but results need expert interpretation

Deep Learning & Reinforcement Learning in Buildings

Beyond classical M, when data is large, patterns are complex, or the system must optimise itself

Deep Learning (DL)

Multi-layer neural networks, learn complex non-linear patterns automatically

LSTM / GRU — sequence modelling, time series prediction
CNN — pattern recognition in images and spatial data
Transformer — long-range dependencies in time series

Building examples: predict indoor temperature 24h ahead · fault detection from sensor patterns · occupancy prediction from image data

Reinforcement Learning (RL)

Agent learns by trial and error, maximizes a reward signal over time

Model Predictive Control (MPC)

Optimal HVAC setpoints, minimize energy while maintaining comfort

Demand response

Learn when to shift loads based on grid signals

Key challenge: training RL in real buildings is risky, simulation environments (EnergyPlus gym) are used first

From Data to Model: The Pipeline

The model is only as good as the features you feed it, domain knowledge is your advantage

1

Feature engineering

Choose & transform inputs
5-day moving avg · time-of-day · building characteristics
Domain knowledge matters most here

2

Train / val / test split

3 sets, not 2
Train: fit model
Validation: tune hyperparameters
Test: final unbiased evaluation

3

Model training

scikit-learn · XGBoost
TensorFlow · PyTorch
Start simple, linear model as baseline

4

Evaluation

MAE · RMSE · R^2
0.5°C MAE vs 2°C MAE context matters
Compare against baseline always

5

Iteration

Tune · select features
Compare models
Document every run

Model Explainability: Why Does the Model Predict That?

A model that works but cannot be explained is hard to trust, publish, or deploy in practice

Why explainability matters in building research:

- Reviewers and clients ask: why does the model say this?
- Unexplained predictions can encode errors in training data
- Physical plausibility check, does solar radiation matter more than outdoor temperature?

SHAP values (SHapley Additive exPlanations):

- How much did each feature contribute to this prediction?
- Works for any model, black box or not
- Python: import shap · shap.summary_plot()

Feature importance, what inputs matter most?

- Physically plausible, validates the model, not just the accuracy metric

Big Data & High-Performance Computing

When your dataset no longer fits in memory, or your model takes days to train on a laptop

When do you need it?

20,000 apartments × hourly data × 4 summers ≈ 700M+ rows
Deep learning training on large datasets
Parametric simulation sweeps (thousands of runs)

Data compression & efficient storage

Parquet format instead of CSV, 10× smaller, faster reads
HDF5 for time series
pandas → Dask for out-of-memory dataframes

Parallel computing in Python

multiprocessing · joblib · Dask
Split dataset across cores,run on all apartments simultaneously
GPU acceleration: CUDA · TensorFlow/PyTorch on GPU

HPC clusters

University cluster
Submit jobs via SLURM, run thousands of simulations in parallel
Cloud: AWS · Google Cloud · Azure for scalable compute

For your own research

where does your data come from, and which approach would you start with?

Statistical analysis · supervised learning · unsupervised · deep learning · RL?

Key Takeaways

DATA → CLEAN → VISUALISE → ANALYSE → MODEL → INSIGHT

01 **Data quality first**

No analysis method fixes dirty data. Cleaning and preprocessing is the most important and most underestimated step.

02 **Always visualise before modelling**

Plots reveal patterns, outliers, and structures that summary statistics miss. seaborn and plotly are your friends.

03 **Statistics before ML**

Understand your data descriptively and relationally before building a model. Skipping this leads to models that surprise you.

04 **Match method to question**

Statistics answers what and why. ML/DL answers what will happen. RL answers what should I do. Choose accordingly.

05 **Explainability is not optional**

A model that works but cannot be explained is hard to publish, trust, or deploy. Use SHAP — check physical plausibility.

06 **Domain knowledge is your advantage**

Feature engineering is where building physicists outperform pure data scientists. Physics-informed features beat brute force.

Thank you for your attention!