UNIVERSITY OF TARTU

Institute of Computer Science

Predictive Control of Air-Source Heat Pump Using Environmental and Market Data

Course project

Anti Konsap, Joonas-Tanel Kessel,

Martin Leissoo, Lauri Lopp

Data Engineering

LTAT.02.007

Table of Contents

Project 1

1. Business Brief

Our main objective is to analyse the energy consumption of an A/C unit under different weather conditions while maintaining an average room temperature of 20°C.

KPIs:

- Average monthly energy consumption
- Average monthly savings

This dataset is primarily intended to help home users optimise their energy usage, as heating and cooling play a significant role in energy consumption in our climate. We have already collected real-world IoT device data from an A/C unit and a temperature sensor located in one of our team members' rooms.

Business questions:

1. How can you use an air-source heat pump to keep your home at 20 °C as efficiently as possible, while taking the outdoor temperature into account?
2. At what outdoor temperatures does the A/C need more energy in the given period?
3. How much do electricity price fluctuations impact costs for running the AC unit?
4. When is it cheaper to pre-heat or pre-cool the home? / Could costs be reduced by pre-heating (above 20 °C) before expensive hours and letting the room slightly below 20 °C during peak price hours?
5. During the coldest days in December 2024 – January 2025, how much more energy did the heat pump need to keep 20 °C indoors?
6. Based on December–January weather and price data, what would be the expected savings?
7. How much of the day is the A/C unit actually running or idling at minimum power consumption? Under what conditions could the pump be turned off completely?

2. Datasets

We plan to combine IoT device data with weather data from Meteo (https://meteo.physic.ut.ee/) and hourly electricity price data from Nord Pool. The latter was obtained from the Elering website (https://estfeed.elering.ee/exchange-prices), as Nordpool does not publicly provide data with hourly granularity. By integrating these datasets, we aim to explore business questions such as when it is most efficient to pre-heat or pre-cool a room to conserve energy and reduce costs, especially when future prices are known.

All datasets are available in our GitHub repository:
https://github.com/LauriLopp/DE_project_2025
along with detailed table descriptions provided in this document below.

- **Home sensors:** indoor temperature, heat pump operation, energy consumption (API connection)
- **Weather data:** outdoor air temperature, humidity, precipitation, solar radiation, air pressure, wind speed and direction
- **Electricity price** (hourly): via API or open dataset

3. Tooling

Environment and deployment

Docker - ensures that our entire data engineering environment runs consistently on any machine. Helps to eliminate dependency and requirements issues. It also simplifies connecting APIs (for accessing IoT data), databases (storing the data), and other services.
Airflow - orchestrates data workflow by creating and managing pipelines from collection to serving. It could be used to get data from an API, transform it with dbt and ingest it into a Postgres database.

Ingestion

PgAdmin and Postgres - PgAdmin provides a GUI to interact with Postgres, making it easier to ingest raw data from CSVs and APIs into our database.

Transformation

dbt (data build tool) - transforms messy raw IoT and weather data with different timestamp formats and missing values into a clean, structured star schema.

Serving

Apache Superset - connects to our Postgres database and enables data visualisation and dashboard options to present the results to stakeholders.

Storage

PostgreSQL - stores both our raw and transformed data, enabling easy access to historical data.

4. Data Architecture

Data is available with a one-day delay, which is more convenient, and then the data is complete for one date and is usable for analysis. In the future, we are looking into near-real-time availability options of the data to use the predictive benefits. The weather dataset does not provide an API, while electricity price data can be accessed through a paid API; however, we use the free CSV download option instead. The IoT data can be accessed via the Home Assistant API.

All data must be synchronised to hourly intervals, as the weather station records measurements every five minutes, while the IoT devices and electricity prices are logged hourly.

All datasets are first stored in their original format. In the staging zone, the data is cleaned and structured through processes such as null value removal, format standardisation, and time zone alignment. See the data flow diagram in the Supplementary Figure S1.

Different datasets use varying timestamp and timezone formats, which must be standardised to ensure proper synchronisation and comparability across all sources.

Example	Source	Format	Standard	Description
2025-09-01 01:00:00	Weather station	YYYY-MM-DD HH:MM:SS	ISO 8601 (variant)	Common in SQL databases and logs
2024-12-07T06:00:00.000Z	IoT devices	YYYY-MM-DDTHH:MM:SS.sssZ	ISO 8601	Full ISO 8601 format with timezone (Z = UTC)
07.12.2024 00:00:00	Elering price data	DD.MM.YYYY HH:MM:SS	Non-standard	European local format, often found in Excel or CSV files

Table: Format standardisation for the timezone

Sensors: API connection via open-source Home Assistant home automation system

Weather data: CSV download from the Weather Station of the Institute of Physics, University of Tartu (https://meteo.physic.ut.ee/)

Electricity price: CSV download from Elering (https://estfeed.elering.ee/exchange-prices)

5. Data Model

Selecting a business process

Business process: operating the air-source heat pump (ASHP) to maintain a constant indoor temperature and minimise the cost of electricity and energy use by measuring the power consumption of the ASHP and indoor temperature, and combining this with publicly available weather data and electricity price information.

Declaring the grain

Event = one measuring event of the IoT devices
Grain = one hourly measurement bucket per device. For each hour interval, the raw IoT samples are aggregated into a single row that contains the average value of the metric (e.g., power, temperature, etc.).
The weather data is available at a finer grain, but the grain is unified with the IoT data.

Dimension tables

The star schema has three fact tables:

- **Dim_time** - stores information about the event time dimension, including additional features for more convenient aggregation (e.g., DayOfWeek, IsWeekend, IsPeakHour, etc.). No SCD handling needed, static dimension.
- **Dim_device** - keeps the information of the working device. The device might change over time. Type 2 SCD handling is used because historical calculations depend on the previous device parameters (e.g. MinPower), and changing them would result in incorrect calculations. Device dimension might also be beneficial if the system expands and more devices are used.
- **Dim_location** - stores information about the device's location, the closest weather station, and the pricing region, as this is important information for calculations that may change over time. Type 2 SCD handling is used because, for example, changing the weather information provider might result in a shift in data and cause anomalies

in analysis that would be difficult to tackle without historical location data.

Fact table

The fact table combines the aforementioned dimensions with the measured values of the IoT system, weather and electricity price.
The full Star Schema design can be found in Supplementary Figure S2.

6. Data Dictionary - Table & column descriptions

In the following chapter, we describe the raw data that is being used in this project. We combine data gathered by personal IoT devices with publicly available weather and electricity stock price datasets.

IOT_data

The table describes all the IoT devices used in the project, including the AC unit device and the indoor temperature sensor.

Entity_id as a string

- sensor.tempniiskuslauaall_temperature
- sensor.ohksoojus_power
- sensor.air_purifier_particulate_matter_2_5
- sensor.air_purifier_particulate_matter_10

State as float

For “sensor.tempniiskuslauaall_temperature”, we expect the value to be of type float, i.e., “19.72”, which represents a temperature in Celsius.

For “sensor.ohksoojus_power”, we expect the value to be of type float, i.e., “639.51”, and it represents Watts (W).

For “sensor.air_purifier_particulate_matter_2_5” and “sensor.air_purifier_particulate_matter_10”, we expect the value to be in float type, ie “1.0177798561111111”, and it represents 2.5 µm and 10 µm particle concentrations, respectively, in micrograms per cubic meter (µg/m^3).

For “sensor.ohksoojus_power”, we expect the value to be of type float, i.e., “639.51”, and it represents Watts (W).

Last_changed in UTC

This column represents the time in UTC format when this IoT and its measurement were taken.

Meteo_physicum_data

The table contains data acquired from https://meteo.physic.ut.ee/ website, where we included the following columns. The Estonian translation in brackets.

Time (Aeg) as Timestamp
Presented in a format of YYYY-MM-DD

Temperature (Temperatuur) in float
Temperature measured in Celsius and in Float format.

Humidity (Õhuniiskus) in float
Measured in percentage.

Air pressure (Õhurõhk) in float
Measured in mm/Hg (millimeters per mercury)

Windspeed (Tuule kiirus) in float
Measured in meters per second (m/s)

Precipitation (Sademed) in float
Measured in millimetres.

UV index (UV indeks) in integer
Measured as a UV index.

Illumination (valgustatus) in float
Measured in lux (lx).

Radiant flux (Kiirgusvoog) in float
Measured in Watts (W).

NP_hourly_rate

Nordpool electricity hourly rate data, where we get the exact price for each hour. The Estonian translation is provided in brackets.

Period (Periood) in UTC
This column represents a time in UTC format: DD.MM.YYYY HH:MM.

Stock price with VAT (Börsihind käibemaksuga) in float
Represents the stock price in cents/kilowatt hour.

Stockprice without VAT (Börsihind ilma käibemaksuta) in float
Represents the stock price in cents/kilowatt hour.

7. Demo Queries

Answers to the first business questions can be found with the queries provided below. Other queries are presented in the file demo_queries.sql.

Q1: Power vs Outdoor Temperature Analysis

```
WITH bucketed AS (
  SELECT
    5*FLOOR(f.OutdoorTemp/5.0) AS bin_start,
    f.ASHP_Power
  FROM FACT_HEATING_ENERGY_USAGE f
  JOIN DIM_DEVICE dd ON f.DeviceKey = dd.DeviceKey
  WHERE f.OutdoorTemp >= -25
    AND f.OutdoorTemp < 30
    AND f.ASHP_Power IS NOT NULL
    AND dd.Model = 'Daikin_123'
)
SELECT
  bin_start,
  bin_start + 5 AS bin_end,
  AVG(ASHP_Power) AS avg_power_w,
  COUNT(*) AS hours
FROM bucketed
GROUP BY bin_start
ORDER BY bin_start;
```

Q2: Data Extraction for Machine Learning

```
SELECT
  dt.FullDate,
  dt.HourOfDay,
  dt.IsWeekend,
  f.OutdoorTemp,
  f.IndoorTemp,
  f.IndoorTemp - f.OutdoorTemp AS TempDelta,
  f.ASHP_Power              AS power_w,
  f.ASHP_Power/1000.0       AS energy_kwh_per_hr
FROM FACT_HEATING_ENERGY_USAGE f
JOIN DIM_TIME dt ON f.TimeKey = dt.TimeKey
JOIN DIM_DEVICE dd ON dd.DeviceKey = f.DeviceKey
```

```
WHERE f.IndoorTemp BETWEEN 15 AND 25
  AND f.OutdoorTemp IS NOT NULL
  AND f.ASHP_Power IS NOT NULL
  AND dd.Model = 'Daikin_123'
ORDER BY dt.FullDate, dt.HourOfDay;
```

Team contribution and LLM usage

Table 2. Workload during the project

Name	Workload
Anti Konsap	1/4
Joonas-Tanel Kessel	1/4
Martin Leissoo	1/4
Lauri Lopp	1/4
Total amount done:	1

LLM usage:
https://chatgpt.com/share/68e2c181-8ae8-800a-9147-ba9c93bc6e8a

Supplementary

Figure S1: Data architecture diagram

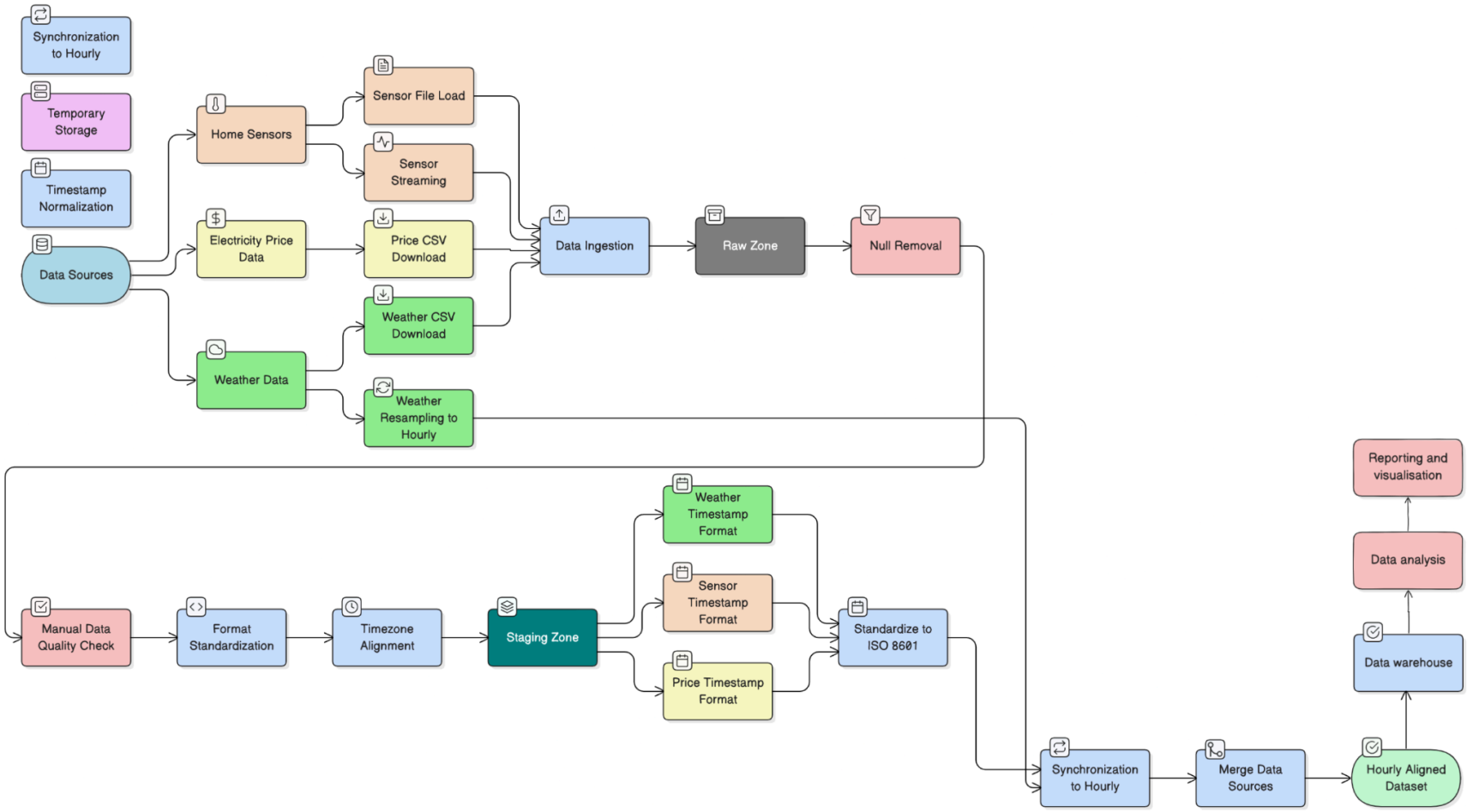

Figure S2: Star Schema design