Graded Assignment 2: Hypothesis Testing
A deeper dive into air quality and the weather

Introduction

Welcome to the second graded assignment of the course Foundations of Data Analytics (2IAB1)! In this assignment, you work in small groups of about 4 students.

In the first (individual) graded assignment, you investigated a particular set of air compounds and weather factors at a given location. In this graded assignment you will formulate and test a hypothesis about a relation between air quality and weather or about air quality patterns supported by the weather data. You will write a report about your work and give a pitch presentation highlighting your results at the end of the project.

To do so, you make use of data obtained from the same sources as in GA1: air quality data from luchtmeetnet (collecting and making available data on the concentration of substances in the air) and weather data from the KNMI.

Before you start

It is important to read (or at least skim through) the entire assignment before you start. Same goes for the rubric, which can be found on Canvas.

This assignment is not only about writing code. Your notebook is *also* a report. If you already know what needs to go in it, it will help you stay organized and finish everything on time.

We also recommend taking a little time to get to know your group members. When you first meet, it could be nice to tell the group a little about yourself and your strengths and weaknesses, so that you may be able to help each other better. You can make up a team name and lay some ground rules during this time as well.

Tutor meetings

You must prepare for these meetings beforehand.

Appoint a chair and a secretary: the chair is responsible for creating an agenda and keeping the meeting on track, while the secretary is responsible for taking notes (called *minutes*) of the meeting. We recommend not only taking minutes during meetings with the tutor, but also during discussions without the tutor.

You must submit the following materials before each meeting:

- your group's Jupyter notebook,
- plans,
- logbook,
- agenda, and
- minutes from the previous meeting

Also have your materials ready to look into during the meeting, especially those you want to discuss.

The agenda typically should consist of the following parts:

1. Work done since the previous meeting: Here you show and tell who did what and your tutor can give feedback and fill in a part of the rubric.
2. Plans for work until the next meeting: Here you show plans that make clear who will be doing what. This is also graded.
3. Questions from the group: Anything else you ran into that you need help with or want feedback on.

An agenda for the first meeting could then look like this:

1. Work done
   (a) Wrote down questions about the chosen hypothesis [together]
   (b) Answered most easy questions [together]
   (c) Investigated a good threshold value for X [Jhin + Leona]
   (d) Investigated if there is enough data to split on Y [Annie]
   (e) Define criteria to determine what is 'better' [Zac]
   (f) Put everything together into final hypothesis [together]
2. Next steps
   (a) <Your plans, clearly listed step by step>
3. Questions
   (a) <Your questions that you ran into before the meeting>

During the meeting

Your tutor will guide your group throughout the project. You will have 15-minute meetings with your tutor during the regular lab session hours. In these meetings you will discuss your progress and the further steps you plan to take, and get feedback and suggestions from your tutor.

Tip: for the best feedback, make sure to come prepared with questions. Which things do you want the tutor to look at and what do you want to know about it? Keep in mind you only have 15 minutes.

During the lab sessions (outside of the tutor meeting) you can work on your programming exercises and/or your assignment. There will still be a tutor or a senior supervisor in the room to answer questions about the exercises. They may also be able to answer some questions about the assignment.

Planning and logbook

From the first day your group comes together (Monday, week 5), you will need to start planning the work of the group and the work distribution within the group. You keep track of this in a *planning document* and you update this planning document during the project. In your planning document you make clear who does what and when, as well as how you organize the work (e.g. where you keep project files). Your time budget for GA2 is about 4 hours per person both in week 5 and in week 6, 14 hours in week 7, and 4 hours in week 8. Keep in mind that a plan is a living document that changes over time. As you progress through the assignment, you will discover more tasks. You must therefore keep your plan up to date.

You must also keep track of the time you spend on this assignment in a *logbook*: for each person it should be clear how much time they spent on which tasks. Be ready to show your planning and the logbook to your tutor during the tutor meetings.

Timeline

The table on page 3 shows the topics and meetings during the lab session hours. Please, plan your work outside the lab session hours together with your group!

Table 1: Timeline of GA2

Week	Contents
5	Monday: • Kick-off for this graded assignment. • Part 1a: Hypothesis selection. Discuss your hypotheses with your tutor **during the lab session!** • Part 1b: Hypothesis refinement. Finish at least step 3 before you leave. Make sure to plan another meeting with your group to put everything together before next week. Thursday: • No tutor meeting. • Programming Exercises DAS and/or work on GA2. **Deadline**: Finish Part 1b: Hypothesis refinement and make a plan and work distribution for Part 2: Queries and data cleaning. Submit all materials for Tutor meetings **before** the meeting.
6	Monday: • Tutor meeting: discuss the finished Part 1b: Hypothesis refinement and planning for Part 2: Queries and data cleaning. • Work on Part 2: Queries and data cleaning. Thursday: • No tutor meeting. • Programming Exercises HYP and/or work on GA2. **Deadline**: Finish Part 2: Queries and data cleaning and make a plan and work distribution for Part 3: Hypothesis testing and interpretation. Submit all materials for Tutor meetings **before** the meeting.
7	Monday: • Tutor meeting: discuss the finished Part 2: Queries and data cleaning and planning for Part 3: Hypothesis testing and interpretation. • Work on Part 3: Hypothesis testing and interpretation. **Deadline**: Finish Part 3: Hypothesis testing and interpretation and make a plan and work distribution for Part 4: Polishing and pitching results. Submit all materials for Tutor meetings **before** the meeting. Thursday: • Tutor meeting: discuss the finished Part 3: Hypothesis testing and interpretation and planning for Part 4: Polishing and pitching results. • Work on Part 4: Polishing and pitching results. **Deadline**: Finish Part 4: Polishing and pitching results. Proof-read final report (Jupyter notebook) and make sure it is complete. Submit your report and slides on Canvas **before** Friday 21:00.
8	Monday • Pitch sessions - **compulsory attendance!** • **Deadline** peer reviews: 13:00. The end!

Jupyter Notebooks: code AND report

You receive two Jupyter notebooks. The notebook named `GA2-template.ipynb` is the one you will work in. Unlike the notebook you obtained for GA1, this notebook only contains section headers. In this notebook you will write both your report and the code required for your work. The other notebook you receive (named `GA2_Help_Code.ipynb`) contains some pieces of code that might be useful to you. You're free to copy code from this notebook to your notebook, if you need it, and edit it as you like.

The group notebook

It is your responsibility to write and test your code in a structured way so that your teammates and your tutor can quickly understand what your code does. You do it in your local copy first, and after testing it, you merge it into the group's notebook. All code from all team members needs to be integrated in this single notebook that you submit on Canvas. To make this easier, we recommend adopting the Style Guide for Python Code (`Style_Guide.pdf`).

It is essential to write comments with your code: first to make clear who is responsible, second to explain what the code does in human language. The comments should be clear for anyone taking this course, not just your group. No more than two people should be responsible for the same bit of code. Everyone should contribute to each part, so everyone should be responsible for some parts of the code.

This notebook doubles as a report. This means you should not just be writing the code and comments about what it does. For the report part, you need to show both creative and critical thinking. That means coming up with interesting hypotheses, well thought out refinements, useful exploration, and good explanations of your reasoning for making certain choices, among other things. Your logic must be correct and your reasoning sensible. Again, everyone should contribute to each part, so everyone should be responsible for some parts of the report.

The help notebook

Some of the code provided in this notebook requires the `geopandas` library. Unfortunately, it doesn't work well with the default environment in Anaconda. To get it running, follow the following steps:

1. Make sure jupyter notebook and Anaconda are not running.
2. Open Anaconda Navigator.
3. Go to Environments (on the left).
4. Click on “Import” (button near the bottom).
5. Click on the folder icon next to “Local drive”.
6. Select “Conda explicit specification files (*.txt)” in the drop-down in the bottom right-hand corner.
7. Navigate to the folder where you extracted the files from the GA2 zip archive.
8. Select `GA2_Geopandas_Spec.txt` and click “Open”.
9. Click “Import”. Now Anaconda will create a new environment with the required packages. This may take some time.
10. Go back to Home (on the left).
11. You can now run jupyter notebook from here as normally. On restarts, make sure your new environment (“geopandas-spec”) selected at the top.

If you run into any issues with this, please ask your tutor for help.

Database structure

You receive a database containing the air quality and weather data. The database is an SQLite database and it contains five main tables:

- `air_quality_stations`,
- `air_quality_data`,

- `weather_stations`,
- `weather_data`, and
- `close_stations`.

Please read the description of the content of each table in Detailed Database Structure on page 15. You can find the names of the table's columns there, as well as their type as defined in SQLite, and their expected contents. You are to decide for yourself which data you need and how to retrieve it using SQL.

Grading

Your assignment will be graded based mainly on the code and report in your group notebook and your pitch. Your group grade also depends on your organization: the tutor meetings, plans, and logbook. Your individual grade may differ from your group grade based on a peer review at the end and input from your tutor based on your presence and participation. This difference may be up to two grade points. For the precise group assessment criteria, see the rubric on Canvas. The peer review is on Feedback Fruits. You can find a copy of the peer review criteria in the same location as the group assessment criteria. Not submitting your peer review may result in a lowered individual grade.

Make sure that the visualizations that you include in your pitch are also produced by your notebook. Double-check if these visualizations are produced correctly on Momotor under the Running submitted notebook tab, i.e., make sure that every visualization in your pitch can also be seen under that tab. Make sure Momotor does not give any errors in any tab. Submit your notebook regularly before the deadline to avoid unpleasant surprises later. If we cannot find your visualizations in Momotor, you will receive no points for the relevant parts. If you are having trouble getting your code to work on Momotor, feel free to ask your tutor or senior supervisor for help.

In case of absence without valid reasons, students may be removed from their groups by senior supervisors. Cases with valid reasons may require compensation work. Being removed from a group is equal to not doing the project, resulting in no grade being awarded.

Part 1a: Hypothesis selection

To start the assignment, you need a problem to solve. In this part, you will formulate several hypotheses to eventually pick one that works.

1. **Share your posters from GA1.**
   One by one, present the poster of your individual GA1 projects to your group. Then, as a group, discuss what parts of the poster you like and/or dislike, mark them on the poster, and write a short comment why. Hand in your posters with your comments to your tutor before you leave.
2. **Write a research question.**
   Having seen the work you each did in GA1, hold a brainstorm session to try and come up with a research question. In case you need some inspiration: you could try to pinpoint major polluters, or find out if there are strong weather influences in particular areas.
3. **Formulate three hypotheses.**
   Now formulate three hypotheses that could help answer the research question. A single hypothesis may only provide a part of the answer, but that is fine. What you need is some hypotheses that could be good candidates for an investigation in this assignment. A good hypothesis for this assignment ...:
   - should be general,
   - should be based either on some observations made in GA1 or on some general knowledge/belief,
   - should lead to potentially useful insights (ask yourself what you could learn depending on the outcomes of your hypothesis testing; how can we use the knowledge you created?),
   - should lead to an interesting result, whether the hypothesis will be rejected or not,
   - should apply to areas (some region of the country or even to the whole country), not specific measuring stations (you will need to use data from multiple AQ *and* weather stations),
   - should use both weather AND air quality data,
   - should not be too easy, nor too hard (if you need help, ask your tutor)*,
   - should need data that is likely to be present in the given data set,
   - should not try to prove a causal relationship.

   *A hypothesis is certainly too easy for this assignment when it requires for example only one scatter plot with linear regression to accept or reject it. This does not become much better by just adding a single extra weather factor or compound.

   An example of a reasonable general hypothesis (unrelated to this project): "Students who do sports perform better in their university studies.". You cannot be sure of the outcome of hypothesis testing in advance, and any outcome, rejecting the hypothesis or not, is interesting and relevant.
4. **Sort your hypotheses.**
   Sort your hypotheses in order of preference, putting the one you would prefer to work on at the top, and write them down in Part 1a: Hypothesis selection of your group notebook as the first part of your report. (Tip: submit your group notebook on Canvas regularly, like now. This is a great way to avoid losing progress as a group if things go wrong. Plus, everyone can easily find it.)
5. **Signal a tutor.**
   Show your three hypotheses to your tutor as soon as you have written them down *during the lab session of Monday week 5.*

   Your tutor will provide you with feedback on which hypothesis has the most potential for this project or will ask you to reformulate one of the hypotheses to make sure it satisfies the requirements.
6. **Finish up your notebook.**
   Write down your approved hypothesis in Part 1a: Hypothesis selection of your group notebook (and submit again on Canvas).

After this part, your group notebook must contain:

- (at least) 3 hypotheses with a brief motivation why you find them interesting, and what you expect the outcome to be (what does it mean?)
- clear indication of which (approved!) hypothesis you will use

Part 1b: Hypothesis refinement

In this part, you will refine the approved hypothesis from Part 1a: Hypothesis selection into one or more testable hypotheses, after which you will make concrete decisions about your measuring stations and your approach in the next part.

1. **Ask questions.**
   Now that you have chosen an approved hypothesis, you need to refine it into one or more testable hypotheses.
   To refine your hypothesis, you need to carefully go over it and note which parts are not sufficiently specific or not testable. Underline any words that could be more specific. For each of the underlined keywords, ask (write down) questions that could help specify this part. Even trivial things are welcome. You do not need to answer these questions right now, that comes in the next step. For every keyword you could ask yourselves questions such as:
   - What is this, exactly?
   - How can we choose a threshold value? Why is that a reasonable choice?
   - Which data represents this keyword? Is that appropriate?
   - Is there enough data to test the hypothesis and to see a baseline?
   - What are confounding variables and how can we reduce the number of confounding variables?
2. **Answer questions.**
   Now start answering the questions from the previous step. Some will be very easy or just need a decision based on common sense. Some will be harder and might need some research and/or discussion. You may also want to check the available data for some of them: when you make decisions you want to make sure you keep enough data. Even when you keep enough data, you still should check that you're not discarding too much. To speed things up, you could do this step in pairs instead of all together.
   You can find an elaborate refinement of the example hypothesis "Students who do sports perform better in their university studies." in Example of a hypothesis refinement.
   Make sure to document your decisions in Part 1b: Hypothesis refinement of your group notebook. Also explain how it relates to the general hypothesis.
3. **Pick measuring stations.**
   After you have refined your hypothesis, look at the map with measuring stations in Fig. 1 (see example code in `Help code.ipynb` to create your own) and decide which ones you want to use. Pick your air quality (AQ) stations, and then the appropriate weather station(s) for each selected AQ station. Not every AQ station is equipped with all sensors, so make sure you pick plenty. Ask yourselves questions like:
   - Does every AQ station have a weather station at a sufficiently small distance?
   - Can you still use the ones that don't? Why?
   - If you can use them, how?
   - Is there any other reason you want to pick or avoid certain stations?

   Document your decisions in Part 1b: Hypothesis refinement in your group notebook.
4. **Design your approach.**
   How do you intend to test these refined hypotheses? Define what needs to be done. Make sure that you have read the complete text of the assignment before you design the approach.
   Your approach must explain:
   - what data (including in which format) you need,
   - where you will get this data from,
   - which technologies you intend to use,

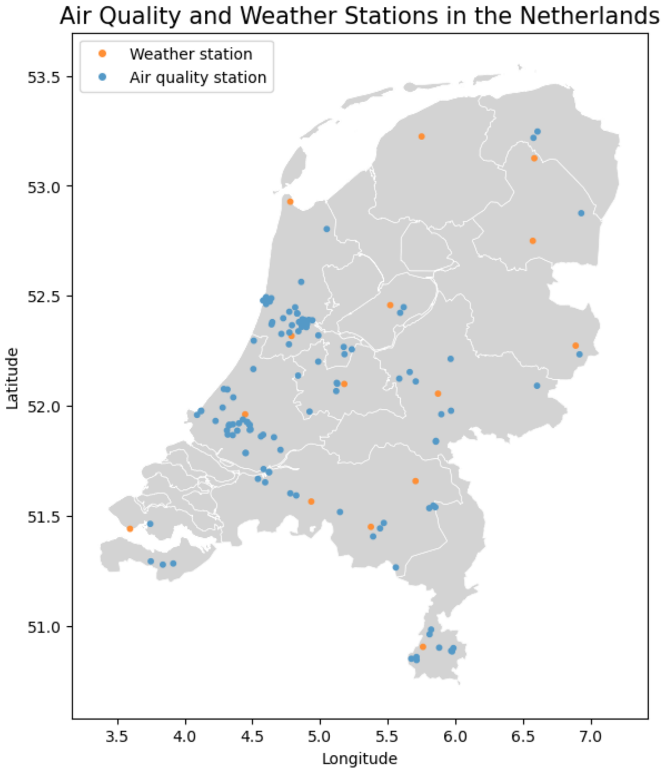

Figure 1: Map of the Netherlands showing locations of Air quality stations and Weather stations.

- any other steps needed to test your hypothesis.

5. **Plan next steps.**
   Make a plan for Part 2: Queries and data cleaning. Who is responsible for which task(s)?

After this part, your group notebook must contain (in addition to the previous parts):

- your refined hypothesis, clearly showing how and why it is different from the original
- a clear indication of which stations you plan to use and why
- your approach, outlining how you plan to go from hypothesis to result

Also submit on Canvas:

- your plans
- your logbook
- your agenda for the first tutor meeting

The assignment continues after the following example.

Example of a hypothesis refinement

Hypothesis: "Students who do sports perform better in their university studies."

Underline keywords

Underline any words that could be more specific:

"Students who do sports perform better in their university studies."

Ask questions

For each of the underlined keywords, ask (write down) questions that could help specify this part. Even trivial things are welcome. You do not need to answer these questions right now, that comes in the next step.

Students

- Which students?
  - Which university/ies?
  - Which degree program(s)?
  - Which period/year(s)?
  - Full-time only or also part-time?
  - Any other requirements?
    - Age?
    - Gender?
    - Health?
    - Relationship?
    - ...

do sports

- Which sports? Does it include e.g. chess or esports? Does it include 1 hour biking to the university every day?
- How often?
  - n *times* per day/week/month/year?
  - x *hours* per day/week/month/year?
- When?
  - At any point during their studies?
  - Specifically during the periods where they are taking courses?

perform better

- What is 'better'?
  - Which one is better:
    - 15 obtained/25 registered ECTS, grades: 8, 8, 8, 4, 2 OR
      15 obtained/15 registered ECTS, grades: 6, 6, 6
    - 15 ECTS, grades: 8, 8, 8 OR
      20 ECTS, grades: 6, 6, 6, 6
  - More ECTS per year?
  - Fewer failed courses per year?
  - Less total time to complete degree?
  - Higher grade average? (how to deal with extra courses when computing averages?)
- Better than what/whom?
  - Students who do not do sports?
  - Correlation more sports ↔ better performance?

university studies

- Any questions relevant here actually already appear with the other keywords, so we can skip this one.

Answer questions

Now start answering the questions from the previous step. Some will be very easy or just need a decision based on common sense. Some will be harder and might need some research and/or discussion. You may also want to check the available data for some of them.

Remember that this is just an example and other answers to the questions could have been chosen. Each keyword could be handled separately, so you might split in pairs and get slightly different-looking discussions.

Students

- Which students?
  - Which university/ies?
    If we have data on all universities in NL, but only data on sports associations in Eindhoven, it doesn't make sense to include all universities.
  - Which degree program(s)?
    If we have sufficient data, we can do the analysis separately per program, and see if there are differences. Otherwise, we can disregard this factor completely.
  - Which period/year(s)?
    We will limit the study to only bachelor students in 3-year programs to get a more homogeneous group. This could eliminate some confounding variables.
  - Full-time only or also part-time?
    Full-time only.
  - Any other requirements?
    - Age?
      Let's limit this from 17-23 to include students that start a bit earlier and also those that finish 1-2 years late. We should check that this does not cost us too much data.
    - Gender?
      It could be a confounding variable. If we have that data, we can split the data on gender; otherwise we ignore gender but be aware that there may be an alternate reason for our observations.
    - Health?
      Not in our data, we cannot take it into account and should report that it could potentially be a confounding variable.
    - Relationship?
      Also not in our data.

do sports

We only have data on membership of sports associations.

We can do some research online to see if we should filter out esports and chess to focus on more traditional physical exercise. It is also possible to investigate differences between these groups.

We cannot include those who do sports outside of these associations. That means that some students labeled as 'not doing sports' actually do sports, and if there *is* a positive correlation between doing sport and study success, then we will have some better-performing sport-doing students mixed in with the worse-performing non-sport-doing students.

We will also have to assume that those who are member of a sports association actually do sports regularly. There may be inactive members. This means that some students labeled as 'doing sports' actually don't do sports, and if there *is* a positive correlation between doing sport and study success, we will have some worse-performing non-sport-doing students mixed in with the better-performing sport-doing students.

We do have data on which years students were members, so we can take students per year and sort them into 2 categories: those who did sports that year, and those who did not.

Alternatively, we should get data from different sources, like student surveys. (Whole different story.)

If we proceed with the data from the university sport associations, we have to reformulate the hypothesis and replace "students who do sports" with "students who are members of a university sport association" and choose answers to the questions we formulated in the same way we did it for answering questions about students.

perform better

We can eliminate some options based on available data, for example if we don't have individual grades. We can also search online for common metrics of student performance and see if we have the data for those. The rest is left open to discussion and a decision needs to be made with the group. Let's assume an online search shows predominantly results in ECTS/year. "perform better" now becomes "earn more ECTS per year".

Write about the hypothesis refinement in your notebook

The original hypothesis was "Students who do sports perform better in their university studies." We have refined it as follows:

- Students who
  - are enrolled in a full-time 3-year bachelor program – to make the population more homogeneous, eliminating some potential confounding variables
  - at TU/e – because we only have data from student sports associations in Eindhoven
  - between 17 and 23 years of age – again homogenizing the population (provided we don't lose too much data)
- and who are members of a university sport association – due to lack of more detailed data
- earn more ECTS that year in their university studies – determined by available data and popular online metrics
- than similar students who are not members of a university sport association for that year. – something to compare against

The refined hypothesis thus becomes: "Students between 17 and 23 years of age enrolled in a full-time 3-year bachelor program at TU/e earn more ECTS in a year in their university studies if they are a member of a university sport association for that year than similar students who are not members of a university sport association that year."

Part 2: Queries and data cleaning

Before you can test your hypothesis, you need to query the database and clean the data you will use. While working on your code, make clear what each code block is supposed to do, especially as you're working on this with multiple people. Here are some tips:

- Consider writing down your agreements somewhere.
- Add comments to each code block: who is responsible for it and what does it do? (This one is **mandatory**.) These comments need to be clear for outsiders and you may not have more than 2 persons responsible per code block.
- Write your code in functions, so you have fewer issues with particular variable names.
- Make clear agreements on how to name functions and variables, so you don't have issues with different parts of the code overwriting previous results for example.
- Consider what parts need to be used multiple times: don't write the same code twice.
- Take a good look at the code provided/written in GA1, some of it may be useful here.
- Make visualizations, mostly for yourself, to check intermediate results: are you indeed working with the correct data?
- Don't forget to keep track of your time spent and changing plans in the relevant documents.

These tips should prove helpful in completing this assignment successfully.

1. **Obtain station codes.**
   In the previous part, you selected on the map which weather and AQ stations you want to use. Now you need to find a way to find the station codes of those stations. Ideally, you do

this by automatically checking for each station if its coordinates are in the area you want. Make a map to show only your selected stations (you can copy the code for the map from `Help code.ipynb` and modify it).

2. **Retrieve data from the database.**
   Each of the keywords in your hypothesis will likely have data associated with it. You need to write queries to retrieve that data from the database for your selected stations. Make sure to inspect your (intermediate) results: is this the correct data? Can you use it in this format?
3. **Clean the data.**
   For example, think of the following potential issues: Are there any unexpected values? Are there any outliers, and what does this mean for your data? Are there any missing values, and what do you do with those? While you write your code, make sure to also write comments on why you are making certain decisions. When merging your code, make sure this all goes into Part 2: Queries and data cleaning of your group notebook.
   Remember that you're working with measurement data from several different stations that were actively measuring in different periods.
4. **Prepare a dataframe for testing.**
   What transformations/aggregations/... need to be done to test your hypothesis? Now you can also merge the data into a single dataframe. Double check that you have everything in there that you need.
5. **Plan next steps.**
   Make a plan for Part 3: Hypothesis testing and interpretation. Who is responsible for which task(s)?

After this part, your group notebook must contain (in addition to the previous parts):

- a list of station codes of the stations that you selected or code to generate such a list
- queries that select the appropriate data from the database
- code for cleaning/transforming/... the data into a usable format
- comments in each code block showing who is responsible (no more than 2 people) and what the code block does

Also submit on Canvas:

- your plans
- your logbook
- your agenda for the next tutor meeting
- your minutes from the previous tutor meeting

Part 3: Hypothesis testing and interpretation

In this part, you apply the skills you learned from the HYP lecture and exercises: Use the data you prepared in the previous part to now test some hypotheses.

1. **What kind of test?**
   What is the exact test you are running? Continue to document your decisions carefully in Part 3: Hypothesis testing and interpretation in your group notebook. The following questions should come to mind:
   - One-sided or two-sided?
   - One-sample or two-sample?
   - Equality of means or equality of proportions?
   - What is your null hypothesis and what is your alternative hypothesis?
   - What level of significance are you using?
   - What assumptions do you (need to) make?

   For all of these questions, you should include a reason why, if the answer is not the only obvious choice. You may also need to run more than one test.
2. **Run the test.**

What is the outcome?

3. **Analyze and interpret the results.**
   Interpret the results in light of your original hypothesis. It is important to not only look at the result of the test, but also to think about what this result means. Can you think of any other reason for the results being the way they are?
4. **Reflect on your hypothesis.**
   Once you have results for your hypotheses, you can now try to argue for or against your original hypothesis. Is it true? False? Or are there other explanations for what you've seen?
5. **Plan the pitch.**
   Make a list of the content you want to include in your pitch. Discuss it with your tutor in the next meeting.
6. **Plan next steps.** Make a plan for Part 4: Polishing and pitching results. Who is responsible for which task(s)?

After this part, your group notebook must contain (in addition to the previous parts):

- a well-executed and well-reasoned hypothesis test (or set of tests) and relevant results
- a thorough interpretation of these results in light of your original (refined) hypothesis
- a concise argument for or against your original hypothesis
- comments in each code block showing who is responsible (no more than 2 people) and what the code block does

Also submit on Canvas:

- your plans
- your logbook
- your agenda for the next tutor meeting
- your minutes from the previous tutor meeting

Part 4: Polishing and pitching results

Your project is nearly complete, but there are still some things to finish up...

1. **Process feedback.**
   Use the feedback you received from your tutor on your work so far to make your analysis better or more complete. For example, you may be able to eliminate some of the other explanations you thought of for your results in Part 3: Hypothesis testing and interpretation.
2. **Prepare the pitch.**
   To finish your project, prepare a pitch in which you present your results. You will have a maximum of 120 seconds to give your pitch to the class. If you go over this time, the teacher will stop you and you may not finish your pitch. Make sure to include your general hypothesis, how you refined it, what your results were, and what this means (for example, what are next steps).
   You may use slides in PDF format. You may use anything you like on your slides, so long as you made it yourself and any plots you use can be found in the notebook submitted on Canvas. We recommend not using more than five slides.
3. **Submit the slides on Canvas on time.**
   Your tutor will combine the slides from all of their groups, so we can make sure the pitch session stays within its time limits. Make sure your tutor has enough time before the pitches by submitting your slides on time.
4. **Give the pitch.**
   We will publish the schedule shortly before the pitch sessions. Attendance is mandatory.
5. **Submit your peer review.**
   After everything is over, we ask you to review the efforts of yourself and your teammates. Fill in and submit the peer review on Feedback Fruits. This is an individual assignment.

Now your notebook should be complete.

- don’t forget to add the required comments with any extra code you wrote for the pitch
- read through the assignment again to make sure you didn’t miss anything

Also submit on Canvas:

- your plans
- your logbook
- your slides for the pitch
- your minutes from the previous tutor meeting
- your peer review (individually)

Detailed Database Structure

Table 2: The `air_quality_stations` table contains data about the stations in charge of collecting air quality metrics and their corresponding measures.

Column	Type	Description
`code`	Text	Code of the air quality station. It is the **primary key** of the table.
`name`	Text	Name of the air quality station.
`latitude`	Real	Latitude at which the air quality station is located.
`longitude`	Real	Longitude at which the air quality station is located.

Table 3: The `air_quality_data` table contains data about the air quality measurements collected at the different air quality stations.

Column	Type	Description
`id`	Integer	Automatically generated identifier of the measurement. It is the **primary key** of the table.
`station_code`	Text	Code of the air quality station where the measurement was collected. It is a **foreign key** pointing to the `code` column of the `air_quality_stations` table.
`datetime`	Timestamp	The Amsterdam winter time when the measurement was collected (UTC+1).
`bc`	Real	Black carbon (C) concentration in μg/m^3.
`co`	Real	Carbon monoxide (CO) concentration in μg/m^3.
`nh_3`	Real	Ammonia (NH_3) concentration in μg/m^3.
`no`	Real	Nitrogen monoxide (NO) concentration in μg/m^3.
`no_2`	Real	Nitrogen dioxide (NO_2) concentration in μg/m^3.
`no_x`	Real	Nitrogen oxides (NO_x) concentration in μg/m^3.
`o_3`	Real	Ground-level ozone (O_3) concentration in μg/m^3.
`pm10`	Real	Particle matter 10, i.e., particles less than 10 µm in diameter (PM10) concentration in μg/m^3.
`pm25`	Real	Particle matter, i.e., particles less than 2.5 µm in diameter (PM2.5) concentration in μg/m^3.
`so_2`	Real	Sulfur dioxide (SO_2) concentration in μg/m^3.

Table 4: The `weather_stations` table contains data about the stations in charge of collecting weather metrics and their corresponding measures.

Column	Type	Description
`code`	Text	Code of the weather station. It is the **primary key** of the table.
`name`	Text	Name of the weather station.
`latitude`	Real	Latitude at which the weather station is located.
`longitude`	Real	Longitude at which the weather station is located.

Table 5: The `close_stations` table contains data about the distance in kilometers between an air quality station and its closest weather station.

Column	Type	Description
`aq_station_code`	Text	Code of the air quality station where the measurement was collected. It is a **foreign key** pointing to the `code` column of the `air_quality_stations` table.
`weather_station_code`	Text	Code of the weather station where the measurement was collected. It is a **foreign key** pointing to the `code` column of the `weather_stations` table.
`distance`	Real	Distance in kilometers between the air quality and weather station.

Table 6: The weather_data table contains data about the weather measurements collected at the different weather stations.

Column	Type	Description
id	Integer	Automatically generated identifier of the measurement. It is the **primary key** of the table.
station_code	Text	Code of the weather station where the measurement was collected. It is a **foreign key** pointing to the code column of the weather_stations table.
datetime	Timestamp	The Amsterdam winter time when the measurement was collected (UTC+1).
wind_direction	Integer	Mean wind direction (in degrees) during the 10-minute period preceding the time of observation (360=north, 90=east, 180=south, 270=west, 0=calm, 990=variable).
wind_speed	Real	Mean wind speed (in 0.1 m/s) during the 10-minute period preceding the time of observation.
wind_gust	Real	Maximum wind gust (in 0.1 m/s) during the hourly division.
temperature	Real	Temperature (in 0.1 degrees Celsius) at 1.50 m above the ground at the time of observation.
sunshine_duration	Real	Sunshine duration (in 0.1 hour) during the hourly division, calculated from global radiation (-1 for <0.05 hour).
global_radiation	Real	Global radiation (in J/cm^2) during the hourly division.
precipitation	Real	Hourly precipitation amount (in 0.1 mm) (-1 for <0.05 mm).
air_pressure	Real	Air pressure (in 0.1 hPa) reduced to mean sea level, at the time of observation.
visibility	Integer	Horizontal visibility at the time of observation (0=less than 100m, 1=100-200m, 2=200-300m,..., 49=4900-5000m, 50=5-6km, 56=6-7km, 57=7-8km, ..., 79=29-30km, 80=30-35km, 81=35-40km,..., 89=more than 70km).
cloud_cover	Real	Cloud cover (in octants), at the time of observation (0=sky completely clear, ..., 4=sky half cloudy, ..., 8=sky completely cloudy, 9=sky invisible).
humidity	Real	Relative atmospheric humidity (in percents) at 1.50 m above the ground at the time of observation.
fog	Integer	Fog 0=no occurrence, 1=occurred during the preceding hour and/or at the time of observation.
rainfall	Integer	Rainfall 0=no occurrence, 1=occurred during the preceding hour and/or at the time of observation.
snow	Integer	Snow 0=no occurrence, 1=occurred during the preceding hour and/or at the time of observation.
thunder	Integer	Thunder 0=no occurrence, 1=occurred during the preceding hour and/or at the time of observation.
ice_formation	Integer	Ice formation 0=no occurrence, 1=occurred during the preceding hour and/or at the time of observation.