Answering Questions with Data

Bridging the gap between technical analysis and stakeholders point-of-view with Jupyter notebooks

Christian Garbin
Dr. Marques' Intro to Data Science - Fall 2020

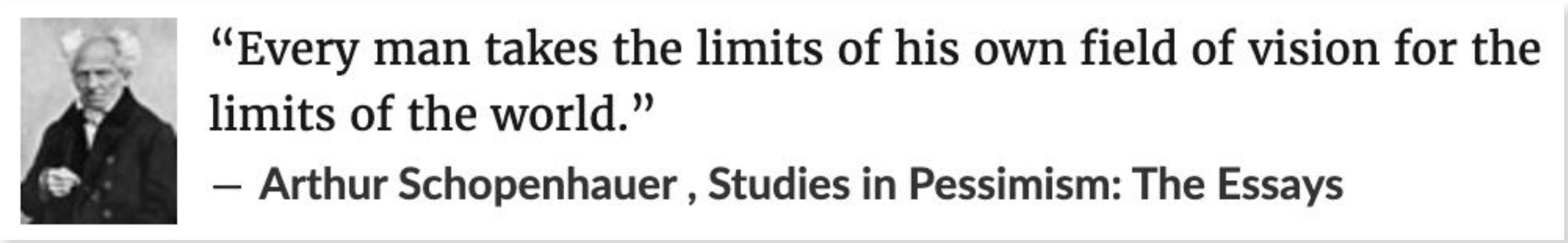
What we will cover today

Using Jupyter notebooks to answer questions

How to write well-structured, understandable, reliable, flexible Jupyter notebooks

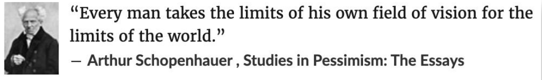
How to present the results of our investigations to the people who asked the questions, the stakeholders

Before we start...



“Every man takes the limits of his own field of vision for the limits of the world.”

― Arthur Schopenhauer , Studies in Pessimism: The Essays



"Every man takes the limits of his own field of vision for the limits of the world."

— Arthur Schopenhauer , Studies in Pessimism: The Essays

My field of vision

- Software engineer, software architect, engineering manager, product owner
- Work on commercial products: must deliver on specific dates, against specific functional and non-functional requirements
- Must support (maintain) what we deliver

Thus, my field of vision, my limits of the world, are the practical implications of technology

I tend to value "how useful this is" over "how interesting this is". It's good for some cases, but not for all cases. For example, I may get results quickly, but I may not explore more creative approaches if I fail to see their usefulness early enough.

Back to what we will cover today...

Using Jupyter notebooks to answer questions

How to write well-structured, easy to understand, reliable, flexible Jupyter notebooks

How to present the results of our investigations to the people who asked the questions, the stakeholders

How we will do it

First we will create a Jupyter notebook to answer a question

Then we will take the results and format them in a presentation for the stakeholders

The example we will use

Question: is there gender discrimination in the salaries of an organization?

Dataset: salaries by gender, age, race, height, and education

Part 1
Creating the Jupyter notebook

Creating the Jupyter notebook

The goal is not only to create a notebook, but to create a **good** notebook

We will start with a working, but not good notebook

We will improve the notebook, step-by-step

What is a **good** notebook?

Overall organization is logical

Important assumptions and decisions are spelled out

Code is easy to understand

Code is flexible (easy to modify)

Code is resilient (hard to break)

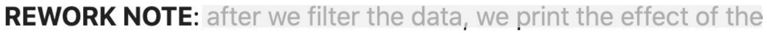
How to follow the stepwise refinement

We will see several notebooks

Each progressively better than the other

Each step has a notebook named …-step-X.ipynb

Notebooks are on [GitHub]



REWORK NOTE: after we filter the data, we print the effect of the

Reworked sections are marked with this text

Switching to the notebooks now…

Step 1: the original notebook, the one that lacks structure and proper coding practices.

Step 2: adds a description, organize into sections, add exploratory data analysis.

Step 3: make data clean-up more explicit, and explain why certain numbers were chosen (the assumptions behind them).

Step 4: make the code more flexible with constants, and make the code more difficult to break (more resilient).

Step 5: make the graphs easier to read.

Step 6: describe the limitations of the conclusion.

…back from the notebooks

Part 2
Presenting the results

Presenting the results - 1

Rule #1: **know your audience**

In this case, our audience is senior management

They want to know if there is gender discrimination, then decide what to do about it

Presenting the results - 2

Rule #2: **choose an appropriate style**

We will use a variation of the inverted pyramid style: start with the conclusion, then present supporting material (if needed)

Why: our audience is interested in the answer, not to the process to get it

Sample presentation starts in the next slide

Gender discrimination in salaries

Sample presentation
to stakeholders

This report investigates the question

> **"Do we have signs of gender discrimination in the company salaries?"**

The answer is a **qualified "yes"**

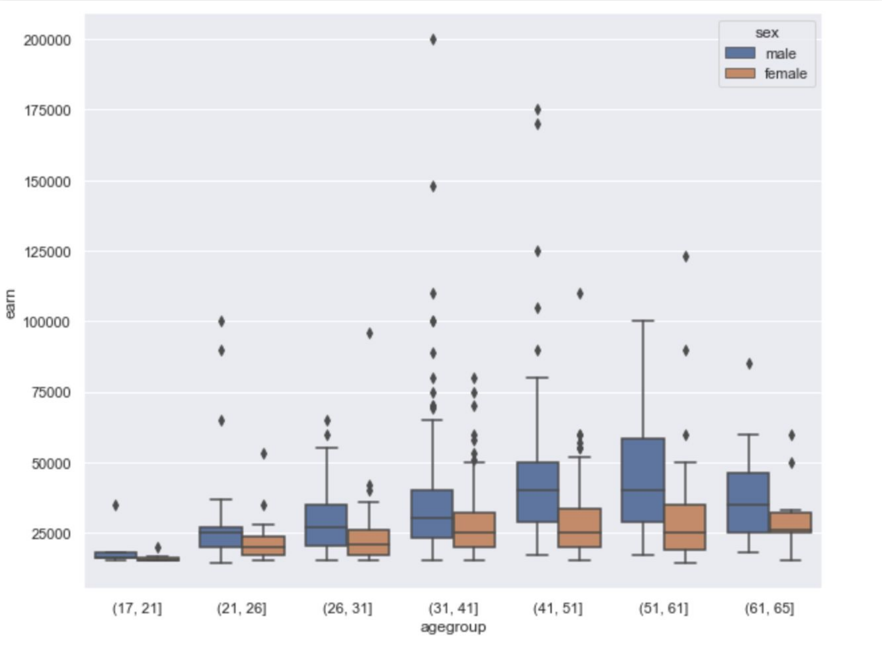
The next slides review the conclusion, the limitations of the conclusion, and recommends actions

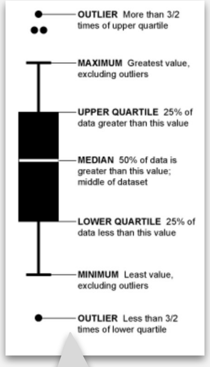
Backup material is available for more detailed discussions

Gender discrimination in salaries

Sample presentation to stakeholders

Salary by age group: females earn less on average. More importantly, females earn less in all quartiles and have fewer outliers in the fourth quartile (the higher end of salaries)





OUTLIER More than 3/2 times of upper quartile

MAXIMUM Greatest value, excluding outliers

UPPER QUARTILE 25% of data greater than this value

MEDIAN 50% of data is greater than this value; middle of dataset

LOWER QUARTILE 25% of data less than this value

MINIMUM Least value, excluding outliers

OUTLIER Less than 3/2 times of lower quartile

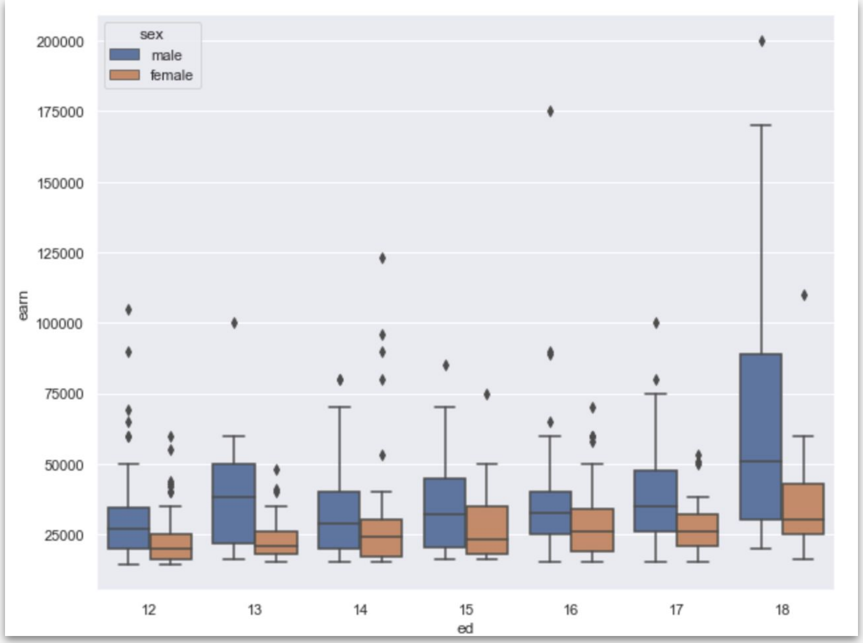
Source

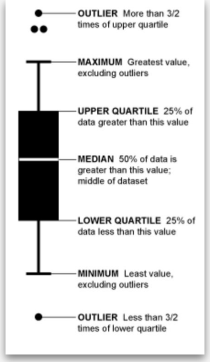
In case not everyone is familiar with boxplots

Gender discrimination in salaries

Sample presentation to stakeholders

Salary by education: females earn less on average, and again the quartiles are lower for females





OUTLIER More than 3/2 times of upper quartile

MAXIMUM Greatest value, excluding outliers

UPPER QUARTILE 25% of data greater than this value

MEDIAN 50% of data is greater than this value; middle of dataset

LOWER QUARTILE 25% of data less than this value

MINIMUM Least value, excluding outliers

OUTLIER Less than 3/2 times of lower quartile

Source

Gender discrimination in salaries

Sample presentation
to stakeholders

Limitation of this analysis: using a proxy data for "same position, same performance"

How the question should have been answered: people in the same position, performing at the same level, should have comparable salaries, independent of their gender

However, the data available does not have the position and performance

How it was answered: we switched to "experience" instead, using age and education as proxies

Gender discrimination in salaries

Sample presentation
to stakeholders

Recommendation

We found strong evidence of gender discrimination, but the data we have is not enough to be conclusive. However, it is enough for the following recommendations:

Recommendation 1: Collect job descriptions and performance evaluation to perform a more accurate analysis.

Recommendation 2: Until that is done, department heads can also review the salaries of males and females doing the same job in those departments and adjust them accordingly.

Although we don't have the exact data we need,
we have enough to recommend actions that can
be take until we get the data we need.

Assumptions made

Sample presentation to stakeholders

1. This report was compiled with a salary list from October of 2020
2. Data was cleaned up to be more representative
   a. Salary must be at least $14,500 - minimum federal wage for full-time employee
   b. At least 12 years of education - at least high-school education to exclude special work arrangements
   c. Not older than 66 years - below the full retirement age to exclude special work arrangements
3. Used age and education as proxies for "experience"

Tip: use numbered lists in reports, so reviewers can say "assumption #3..." instead of "the fifth assumption from the bottom ..."

End of the sample presentation

And that's it for today!

Well, almost...

Some cheating was involved...

It's not always a linear process, like we did here

It's common to go back and forth in the notebook, adding and removing sections, moving cell arounds, reworking section as new information unfolds, ...

Don't be concerned if the first version doesn't look great - keep refining it - **the important thing is to get started**

Some references

Coding

Code Complete 2 (McConnell): write good code in general

Effective Python (Slatkin): write good Python code

Presenting information

Storytelling with Data (Knaflic): basic text on data visualization - start here if you know nothing about presenting data

How Charts Lie (Cairo): more advanced topics on getting charts right

Ok, now that's really it for today!

Quick recap

Notebooks

- Organize in sections
- Spell out assumptions and limitations
- Make code flexible
- Make code difficult to break

Presentations

- Know your audience
- Choose a style - prefer to be brief
- Spell out assumptions and limitations
- Have support material ready

And don't be discouraged if you have to rework again and again until it looks right

One more thing...

Keeping track of the work

Add Git tags (versions) to "freeze" the work