Better Care and Feeding of Machine Learning Models

Towards reproducibility and transparency

AI SINGAPORE

Talk Structure

- Personal reproducibility disaster story
- What is reproducibility?
- Why does it matter?
- Python tools for reproducible workflows
  - Demo 1: cookiecutterML
  - Demo 2: push-button analysis
  - Demo 3: interpretability
- Downstream benefits : interpretability, evolvability

AI SINGAPORE

Me one year ago

- Working hard
- But unorganised folders
- Non-readable filenames
- No version control or README.md
- No tracking of Python libraries and environments used

AI SINGAPORE

collaborators not impressed

AI SINGAPORE

In fact my project folder turned into an entire workshop on reproducibility

AI SINGAPORE

How I felt...

AI SINGAPORE

Future me also not impressed

AI SINGAPORE

Why reproducibility?

Your collaborators will thank you

Future self will thank you

Reduce technical debt

Hedonic Housing Prices and the Demand for Clean Air

Daniel L. Rubinfeld
David Harrison, Jr., *Harvard University*

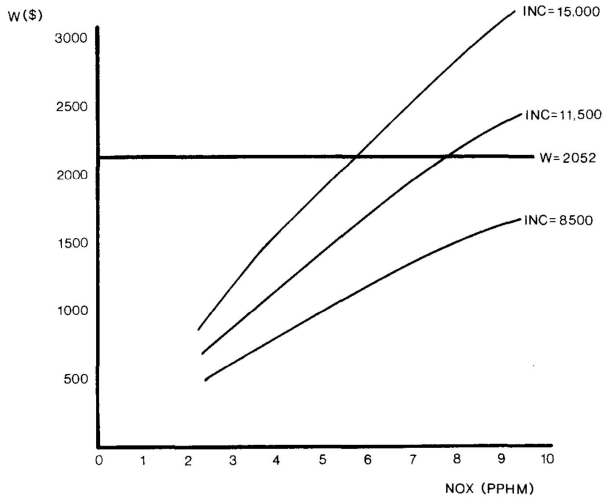
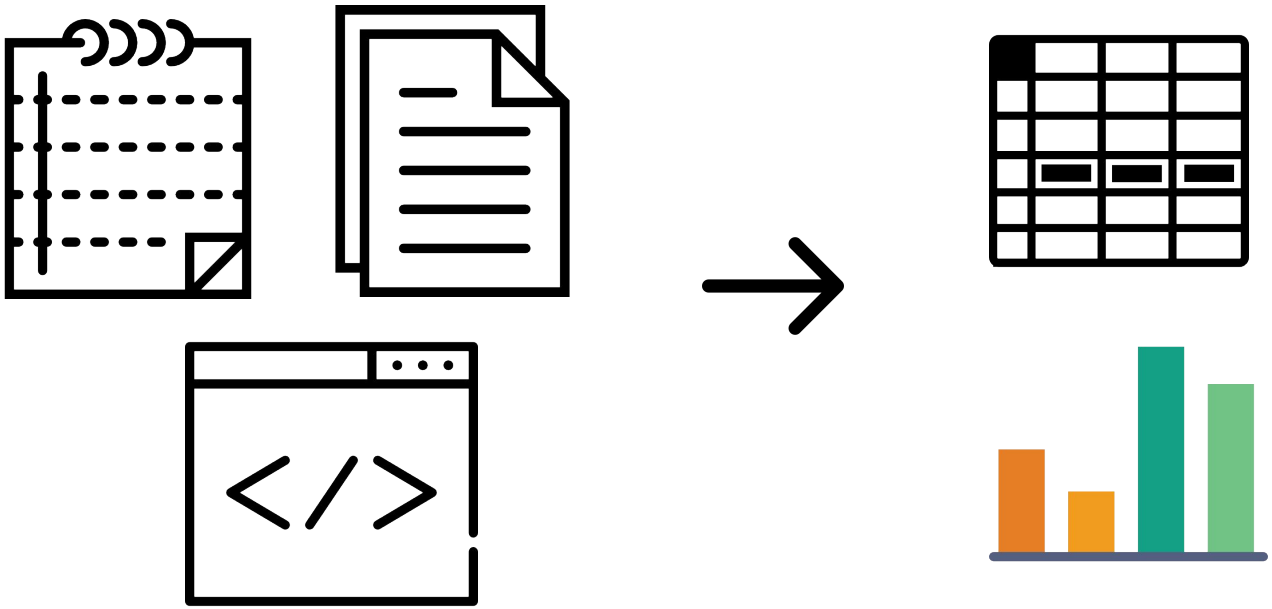


FIG. 1. Willingness to pay for 1 pphm improvement in *NOX* concentration, by *NOX* for households in three income levels (log–log version).

AI SINGAPORE

Computational Reproducibility



https://www.practicereproducibleresearch.org/core-chapters/3-basic.html

Images: flaticon.com

AI SINGAPORE

Reproducibility has different levels

complexity

Well-structured folders

Searchable filenames

Functions and automation with scripts and build tools

Reproducible Reports

Version control

Standardised environments

Containerisation

Some quick simple wins here.

AI SINGAPORE

sources

1. Well-structured folders http://kbroman.org/steps2rr/pages/organize.html
2. Searchable filenames http://www2.stat.duke.edu/~rcs46/lectures_2015/01-markdown-git/slides/naming-slides/naming-slides.pdf
3. Functions and Scripting http://kbroman.org/steps2rr/pages/scripts.html http://kbroman.org/steps2rr/pages/automate.html
4. Reproducible Reports https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks
5. Version Control Bryan J. (2017) Excuse me, do you have a moment to talk about version control? *PeerJ Preprints* 5:e3159v2 https://doi.org/10.7287/peerj.preprints.3159v2
6. Using Python virtualenv https://realpython.com/python-virtual-environments-a-primer/
7. Introduction to Docker Containers https://docs.docker.com/get-started/
8. Using Docker containers for Reproducible Data Science http://docs.pachyderm.io/en/v1.2.1/examples/word_count/README.html
9. Opinionated Analysis Development Parker H. (2017) Opinionated analysis development. *PeerJ Preprints* 5:e3210v1https://doi.org/10.7287/peerj.preprints.3210v1
10. Examples: https://www.practicereproducibleresearch.org

AI SINGAPORE

Project templates = simple but a big win

Reproducible ML for Minimalists

Pared-down project template for reproducible Machine Learning. Adopted from Cookiecutter Data Science and Mario Krapp

Directory Structure

```
.
├── AUTHORS.md
├── LICENSE
├── README.md
├── models  <- compiled model .pkl or HDFS or .pb format
├── config  <- any configuration files
├── data
│   ├── interim <- data in intermediate processing stage
│   ├── processed <- data after all preprocessing has been done
│   └── raw <- original unmodified data acting as source of truth and provenance
├── docs  <- usage documentation or reference papers
├── notebooks <- jupyter notebooks for exploratory analysis and explanation
├── reports <- generated project artefacts eg. visualisations or tables
│   └── figures
└── src
    ├── data-proc <- scripts for processing data eg. transformations, dataset merges etc.
    ├── viz  <- scripts for visualisation during EDA, modelling, error analysis etc.
    ├── modeling    <- scripts for generating models
|--- environment.yml <- file with libraries and library versions for recreating the analysis environment
```

AI SINGAPORE

Push button analysis demo with housing data

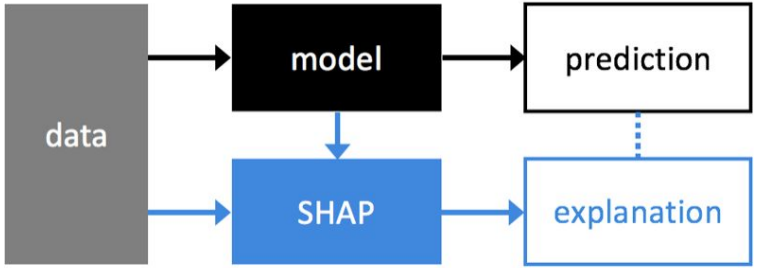
delete all artefacts and recreate entire project using ONLY scripts!

AI SINGAPORE

Our own work - data diffs on top of code diffs

AI SINGAPORE

Benefits of Reproducible and Transparent models: Interpretability

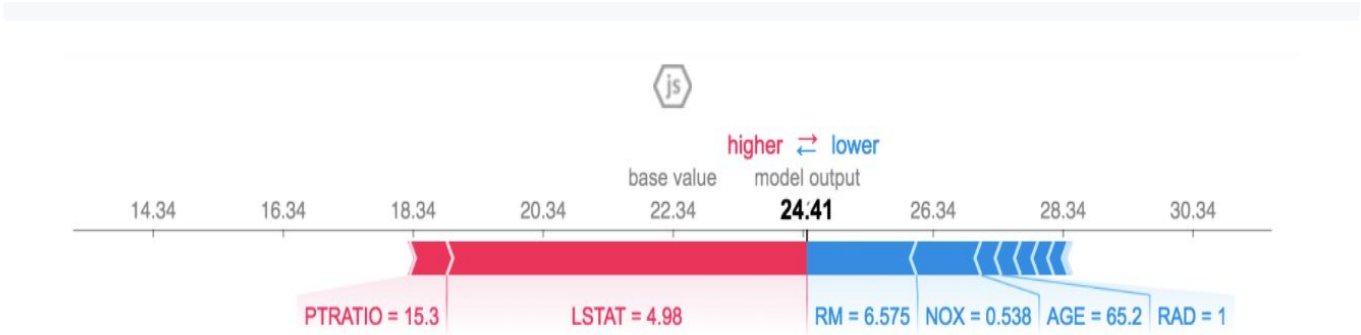


build passing

SHAP (SHapley Additive exPlanations) is a unified approach to explain the output of any machine learning model. SHAP connects game theory with local explanations, uniting several previous methods [1-7] and representing the only possible consistent and locally accurate additive feature attribution method based on expectations (see the SHAP NIPS paper for details).

AI SINGAPORE

Benefits of Reproducible and Transparent models: Interpretability



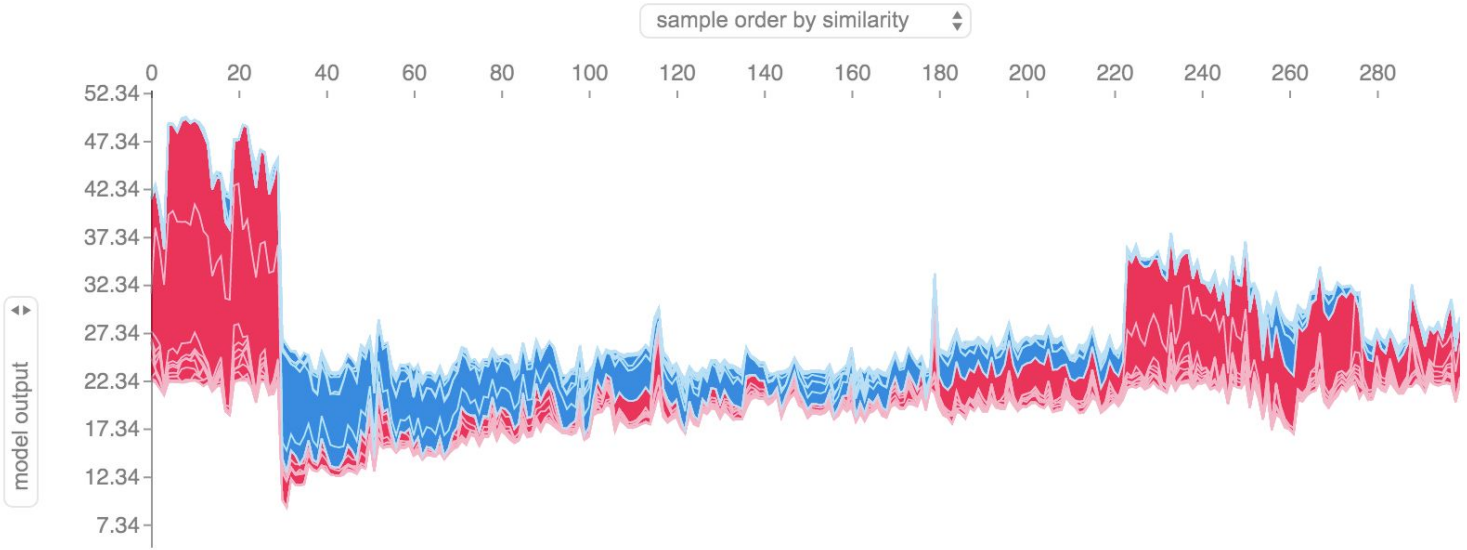
The above explanation shows features each contributing to push the model output from the base value (the average model output over the training dataset we passed) to the model output. Features pushing the prediction higher are shown in red, those pushing the prediction lower are in blue.

resources: https://github.com/slundberg/shap

AI SINGAPORE

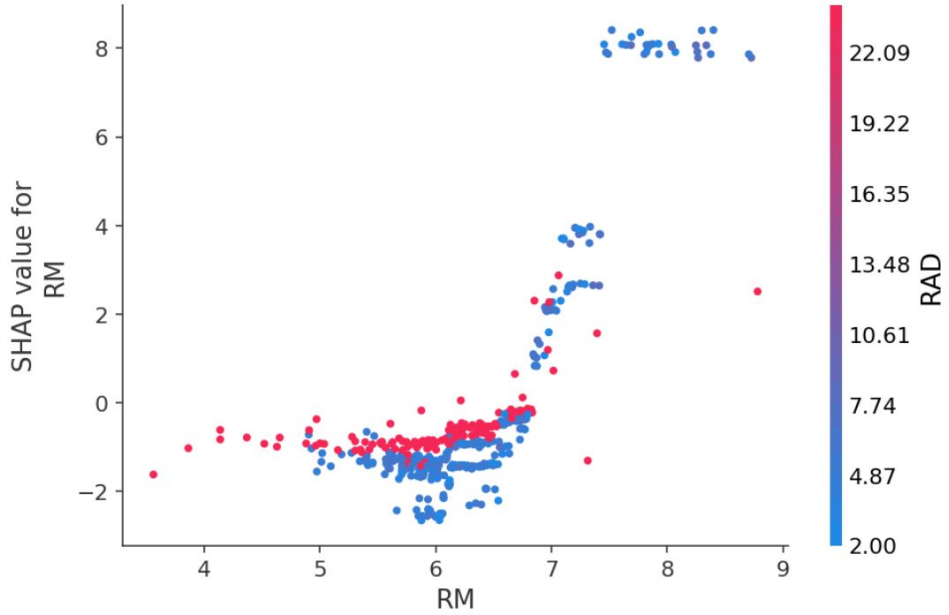
Benefits of Reproducible and Transparent models: Interpretability

```
# visualize the training set predictions
shap.force_plot(shap_values, X)
```



AI SINGAPORE

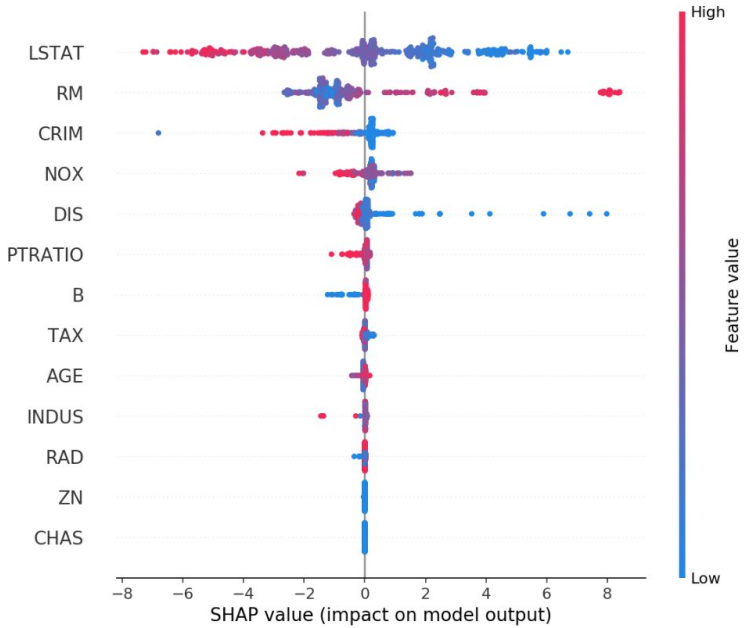
```
# create a SHAP dependence plot to show the effect of a single feature across the whole dataset
shap.dependence_plot("RM", shap_values, X)
```



AI SINGAPORE

To get an overview of which features are most important for a model we can plot the SHAP values of every feature for every sample. The plot below sorts features by the sum of SHAP value magnitudes over all samples, and uses SHAP values to show the distribution of the impacts each feature has on the model output. The color represents the feature value (red high, blue low). This reveals for example that a high LSTAT (% lower status of the population) lowers the predicted home price.

```
# summarize the effects of all the features
shap.summary_plot(shap_values, X)
```



AI SINGAPORE

Other benefits we are interested in

Containers == mix-and-match, manage and track models in the wild

Industry Innovation aka.
Practical Machine Learning

Pushing boundaries of implementation rather than research

https://github.com/jeannefukumaru/pycon-apac2018
https://github.com/jeannefukumaru/cookiecutter-ml

Better Care and Feeding of Machine Learning Models

Towards reproducibility and transparency

AI SINGAPORE