#WhatIfTool #SHAP

HANDS-ON TUTORIAL

Probing ML Models for Fairness With the What-If Tool & SHAP

James Wexler & Andrew Zaldivar
With Sara Robinson, Mahima Pushkarna & Tolga Bolukbasi

https://whatif-tool.dev
https://pair-code.github.io/what-if-tool/fat2020.html

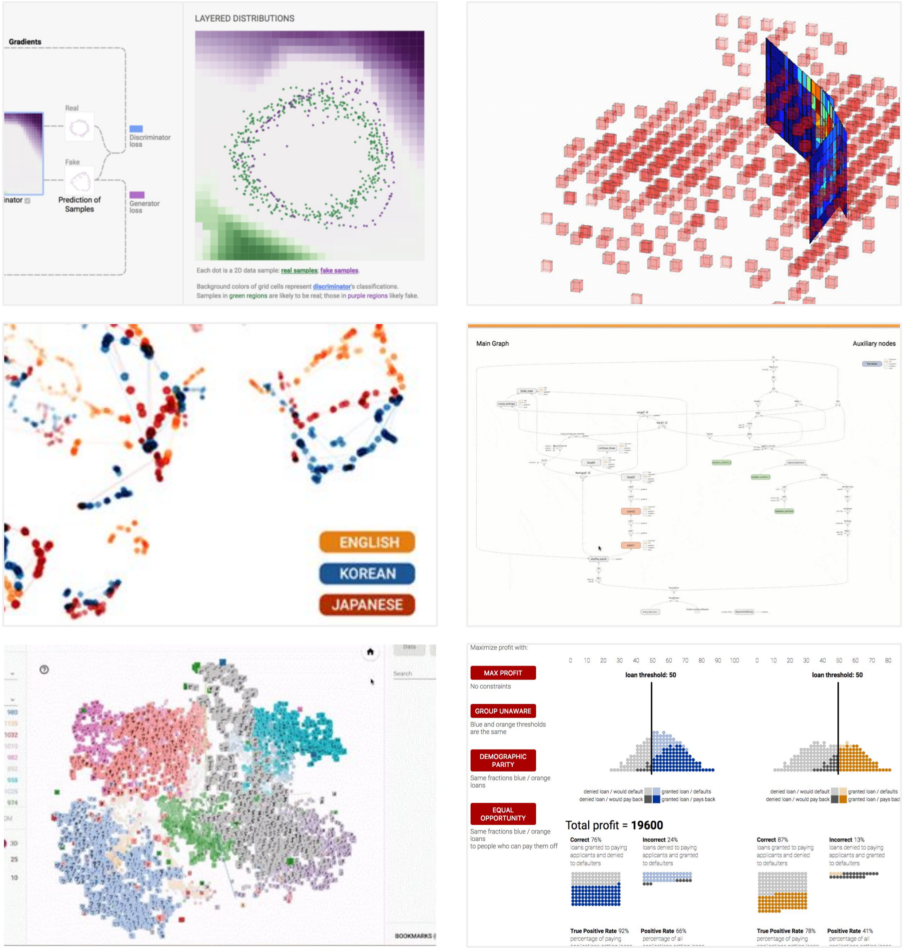
Google

#WhatIfTool #SHAP

Probing ML models for fairness with the What-If Tool & SHAP

PAIR’s mission is to conduct **human-centered research and design** to make **human-AI partnerships productive, enjoyable, and fair.**

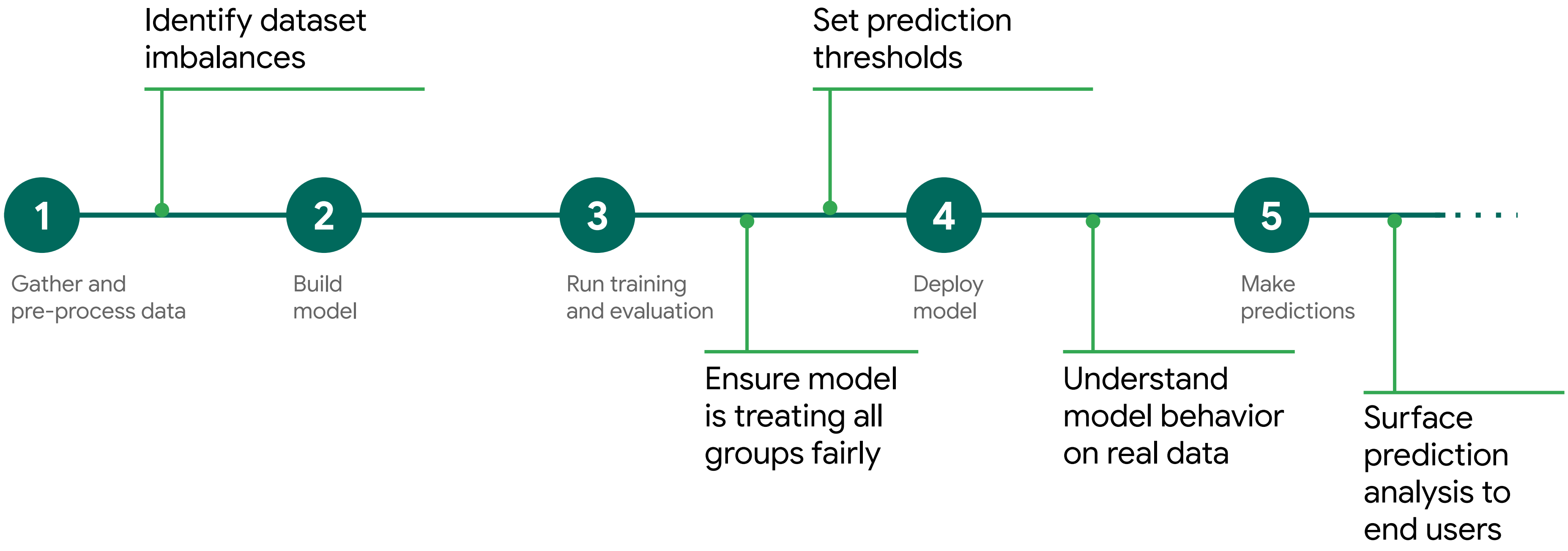
We make technology.



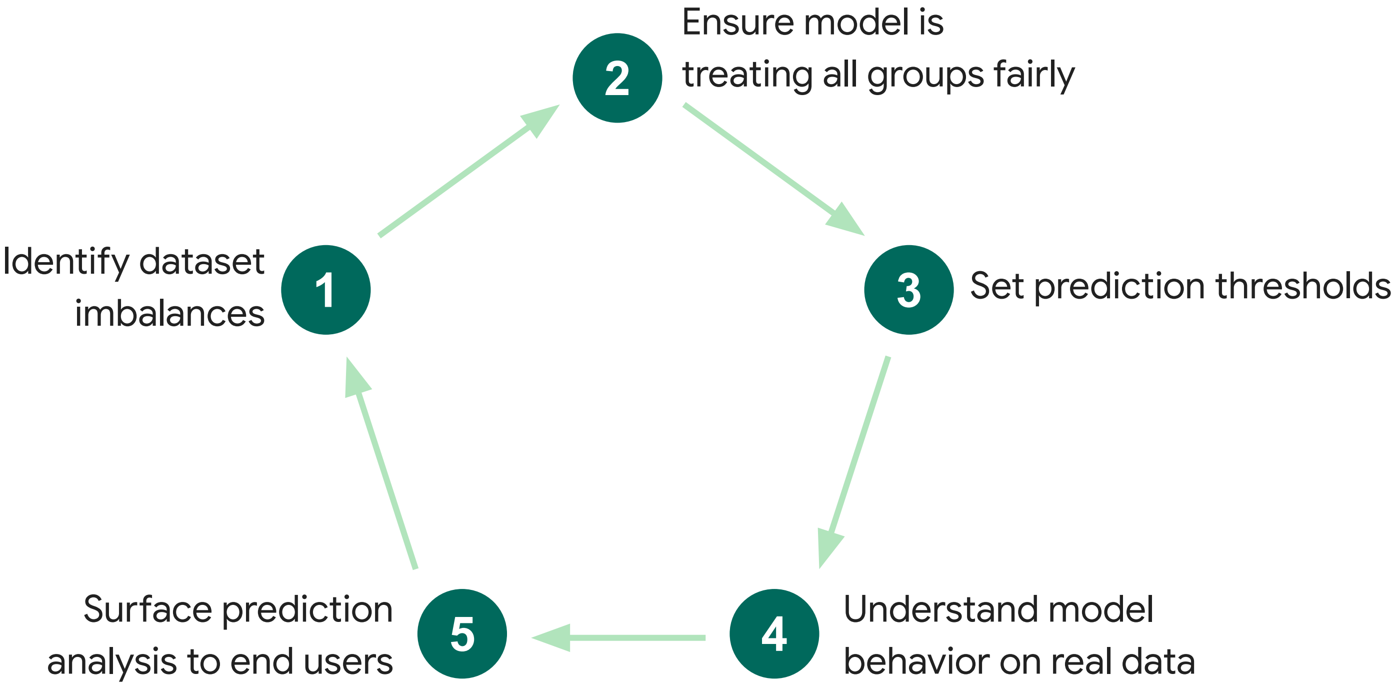
Above: A sampling of work from PAIR.

PAIR

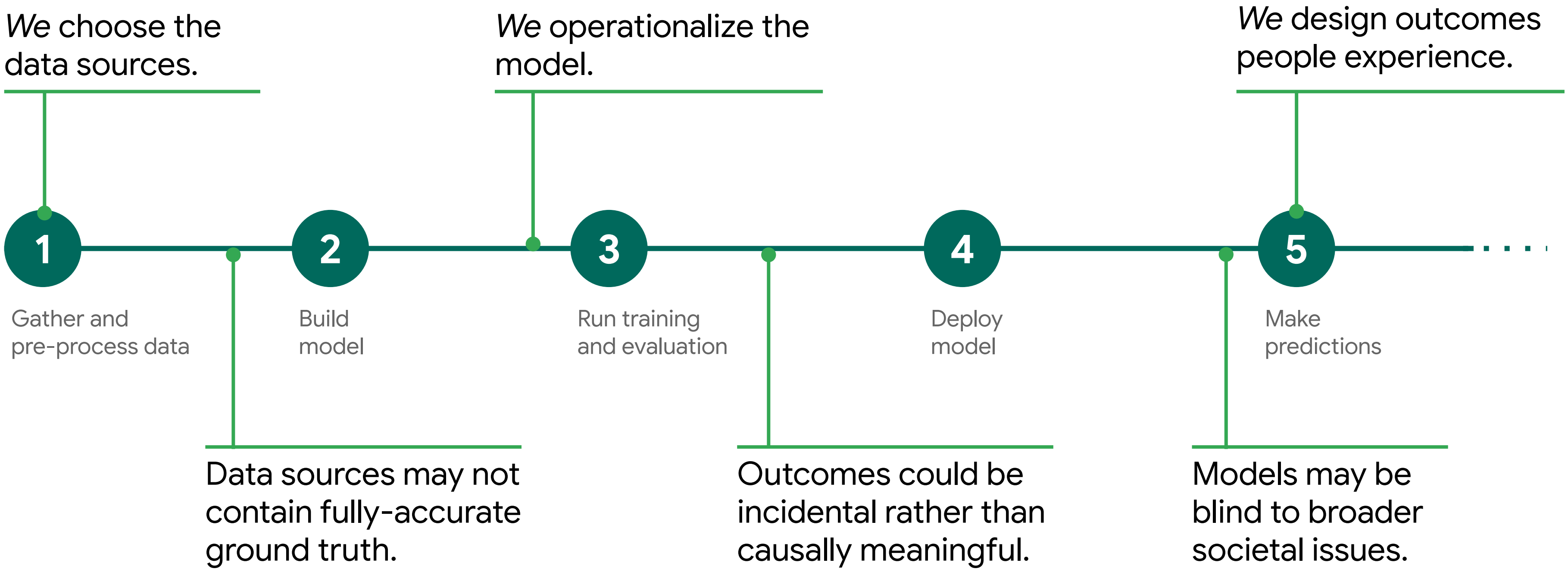
Q: How does **explainability fit into the ML lifecycle?**



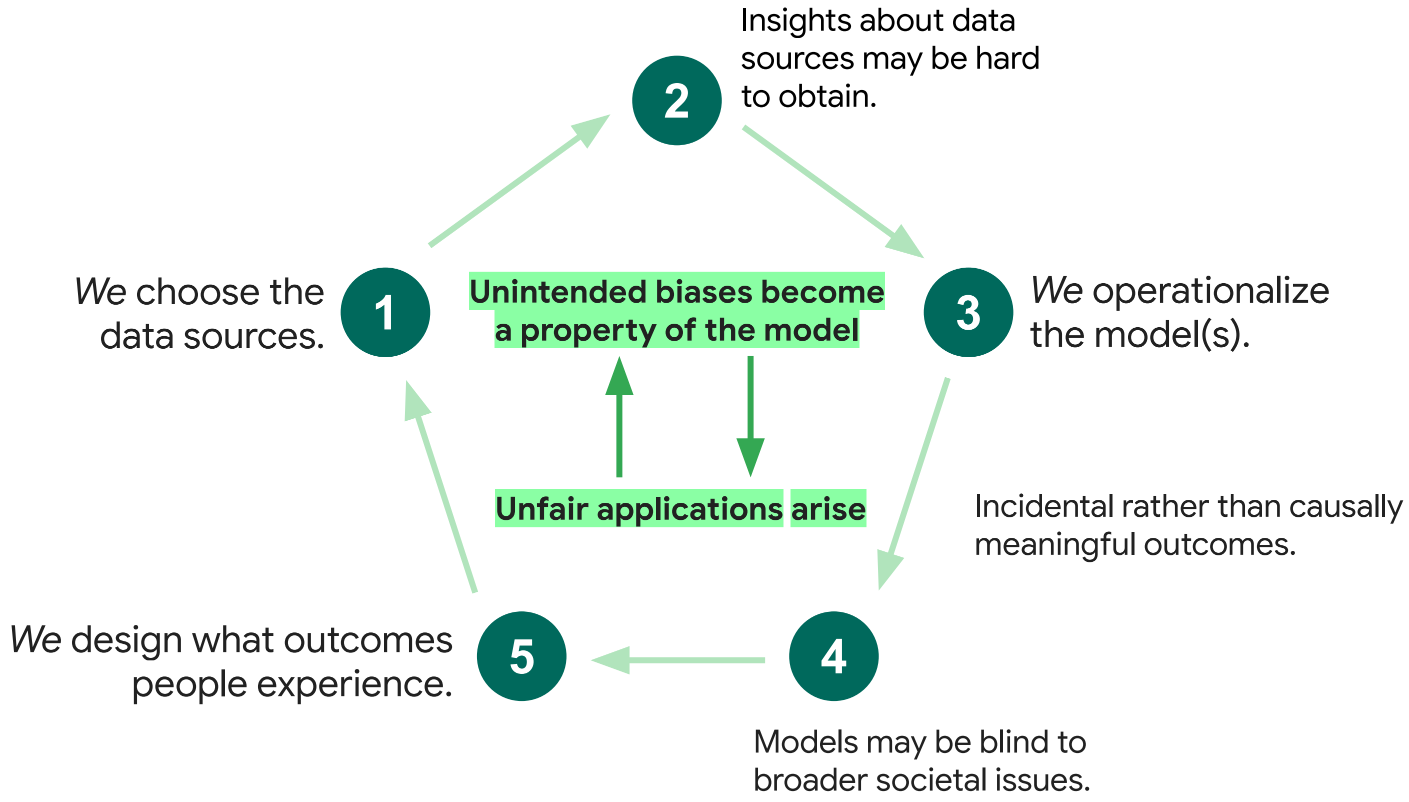
The **explainability process**



Q: How can **biases be introduced into ML lifecycles?**



The *unfairness* process



Algorithmic Unfairness: Some examples

Representational Harm

When an ML system amplifies or reflects negative stereotypes about particular groups.

Opportunity Denial

When an ML system negative impacts individuals' access to opportunities, resources, and overall quality of life.

Disproportionate Failure

When the experience of interacting with an ML system is disproportionately failing for particular groups.

Adapted from "The Trouble with Bias" by Kate Crawford – [NeurIPS 2017 Keynote] (Video)

Google's AI Principles

1. Be socially beneficial.

2. Avoid creating or reinforcing unfair bias.

3. Be built and tested for safety.

4. Be accountable to people.

5. Incorporate privacy design principles.

6. Uphold high standards of scientific excellence.

7. Be made available for uses that accord with these principles.

There are **many different interpretability approaches...**

Feature Attributions	Model & Data Analysis	Gradient & Concept Testing	Datapoint Inspection
Integrated gradients	What-If Tool	TCAV	Partial Dependence Plots
SHAP	Facets	Grad-CAM	Counterfactuals
LIME	Fairness Indicators	Guided Backpropagation	Ablation testing
XRAI			

We’ll focus on **these two:**

Feature Attributions	Model & Data Analysis	Gradient & Concept Testing	Datapoint Inspection
Integrated gradients	**What-If Tool**	TCAV	Partial Dependence Plots
SHAP	Facets	Grad-CAM	Counterfactuals
LIME	Fairness Indicators	Guided Backpropagation	Ablation testing
XRAI			

How **does** my model perform...

classification accuracy / precision-recall curve / logarithmic loss / area under the curve / mean squared error / mean absolute error / F1 score / standard deviation / variance / confidence intervals / KL divergence / false positive rate / false negative rate / <insert **metric** here>

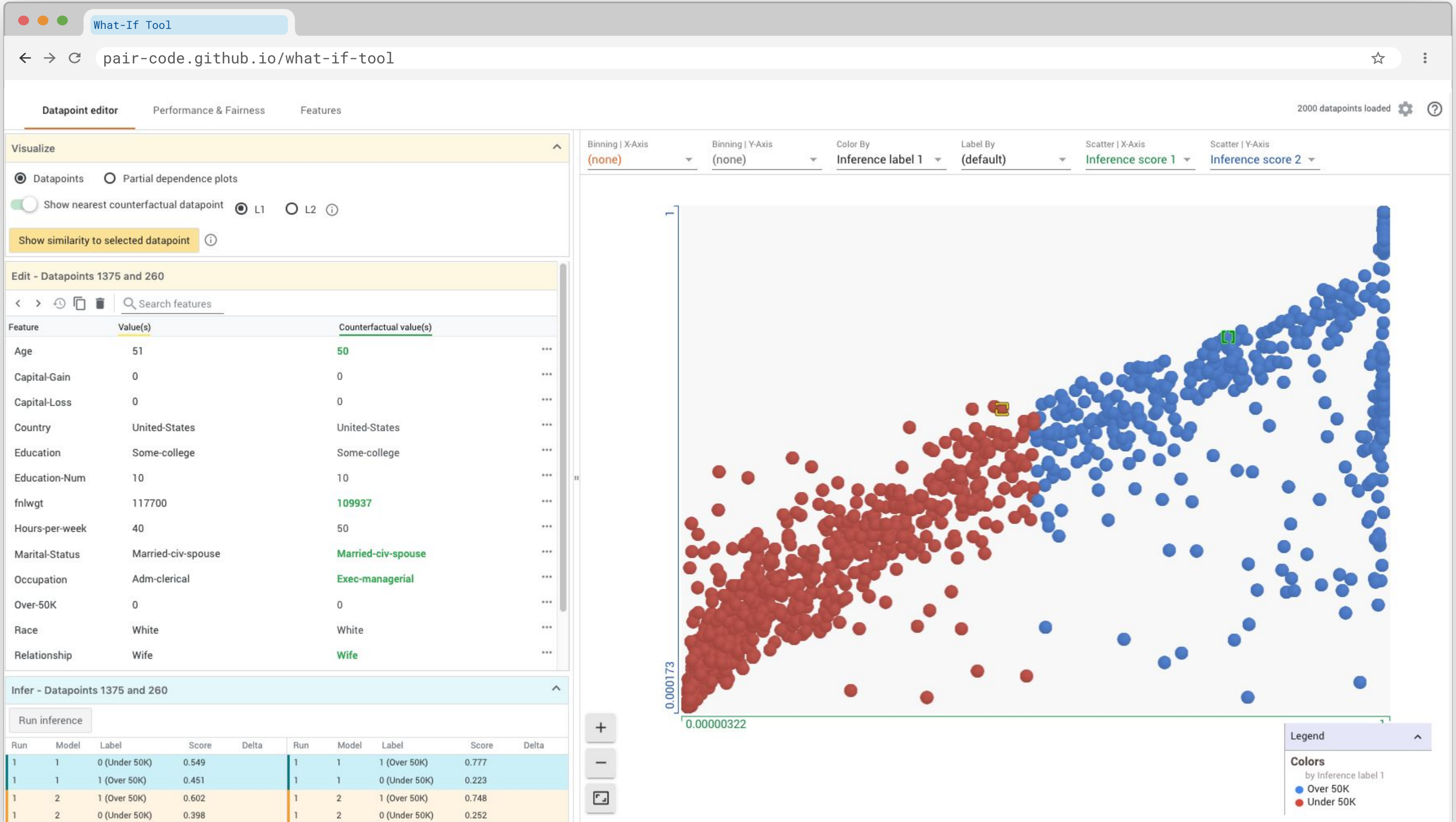
How **might** my model perform...

on subgroups in test data / on cross-slices in test data / on an individual data point / if a datapoint is perturbed / if model thresholds were different/ if optimized differently / across all values of a feature / when compared to a different model / on different data points within a neighborhood of data points /

<insert **question** here>

What if...

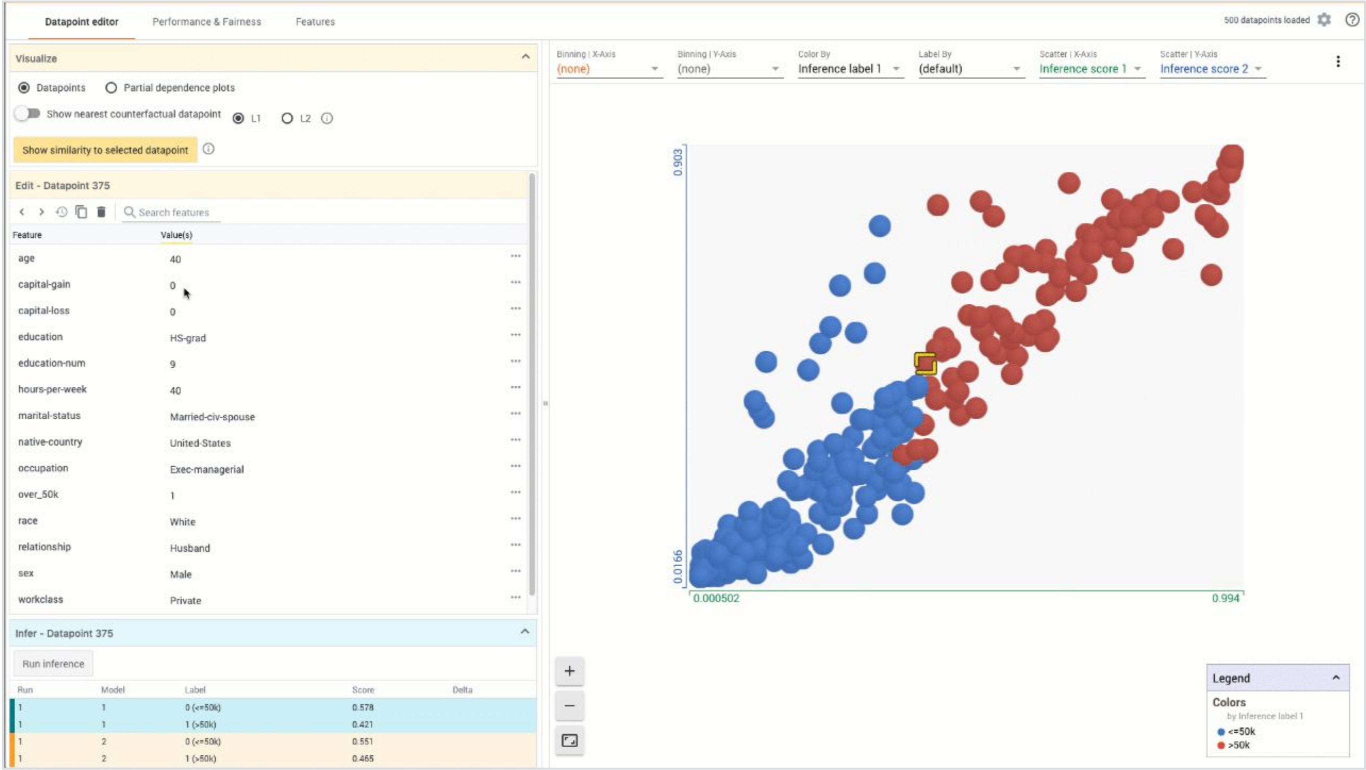
you could inspect
machine learning models,
with **minimal coding**
required?


What-If Tool
pair-code.github.io/what-if-tool
Datapoint editor
Performance & Fairness
Features
2000 datapoints loaded
Visualize
Datapoints
Partial dependence plots
Show nearest counterfactual datapoint
L1
L2
Show similarity to selected datapoint
Edit - Datapoints 1375 and 260
Search features
Feature
Value(s)
Counterfactual value(s)
Age 51 50
Capital-Gain 0 0
Capital-Loss 0 0
Country United-States United-States
Education Some-college Some-college
Education-Num 10 10
fnlwgt 117700 109937
Hours-per-week 40 50
Marital-Status Married-civ-spouse Married-civ-spouse
Occupation Adm-clerical Exec-managerial
Over-50K 0 0
Race White White
Relationship Wife Wife
Infer - Datapoints 1375 and 260
Run inference
Run Model Label Score Delta
1 1 0 (Under 50K) 0.549
1 1 1 (Over 50K) 0.451
1 2 1 (Over 50K) 0.602
1 2 0 (Under 50K) 0.398
Run Model Label Score Delta
1 1 1 (Over 50K) 0.777
1 1 0 (Under 50K) 0.223
1 2 1 (Over 50K) 0.748
1 2 0 (Under 50K) 0.252
Binning | X-Axis
(none)
Binning | Y-Axis
(none)
Color By
Inference label 1
Label By
(default)
Scatter | X-Axis
Inference score 1
Scatter | Y-Axis
Inference score 2
1
0.000173
0.00000322
1
Legend
Colors
by Inference label 1
Over 50K
Under 50K

Supports *What-If* analysis

Easily ask hypotheticals

Alter datapoints
and see how model
outputs change



#WhatIfTool whatif-tool.dev

Above: Editing a feature value and then running inference preserves a history of inference values. Alternatively, users can explore partial dependence plots for each feature, sorted by interestingness.

Counterfactuals

"What would have to change for me to have gotten the loan?"

$$\arg\min_{x'} \max_{\lambda} \lambda(f_w(x') - y')^2 + d(x_i, x')$$

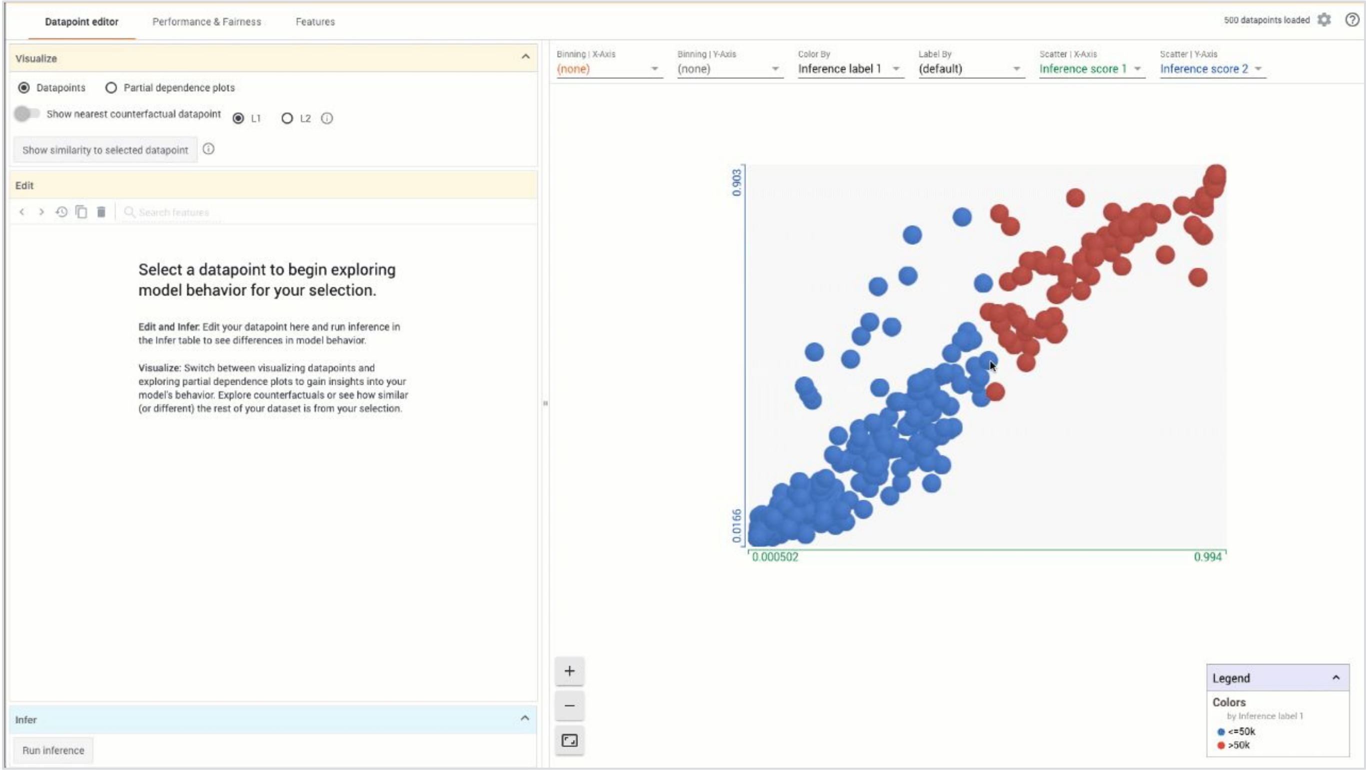
Approaches

- Optimization problem to find hypothetical datapoint
- Search across real examples

--
Wachter et al. "Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR"

Explore decision boundaries

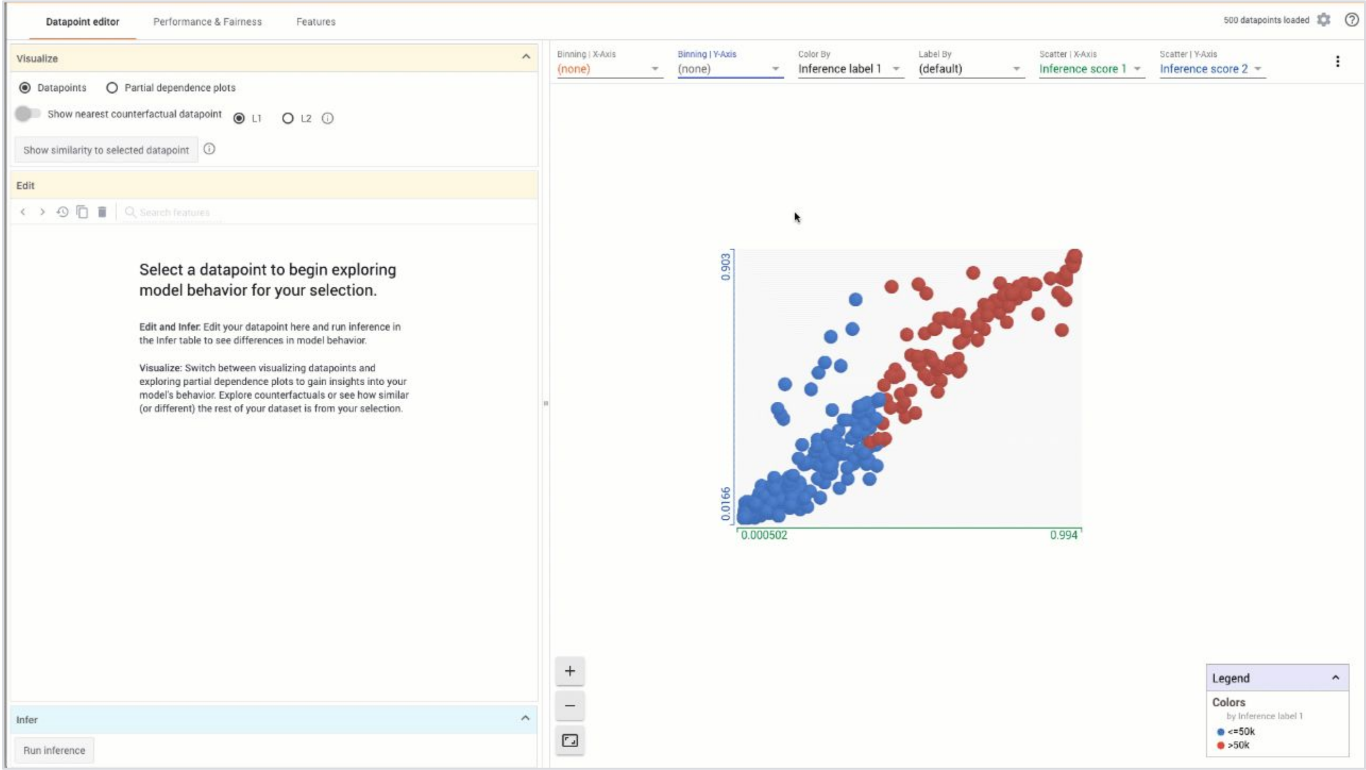
Translating counterfactual research into visual tooling within workflows.



Above: For classification problems, our counterfactual finding feature can identify the most similar datapoint (to a selection) in the loaded data that was classified differently by the model. For any dataset, L1 & L2 distances are available as inbuilt similarity metrics. However, users can specify custom metrics when invoking the tool.

Scale up without changing user's mental models

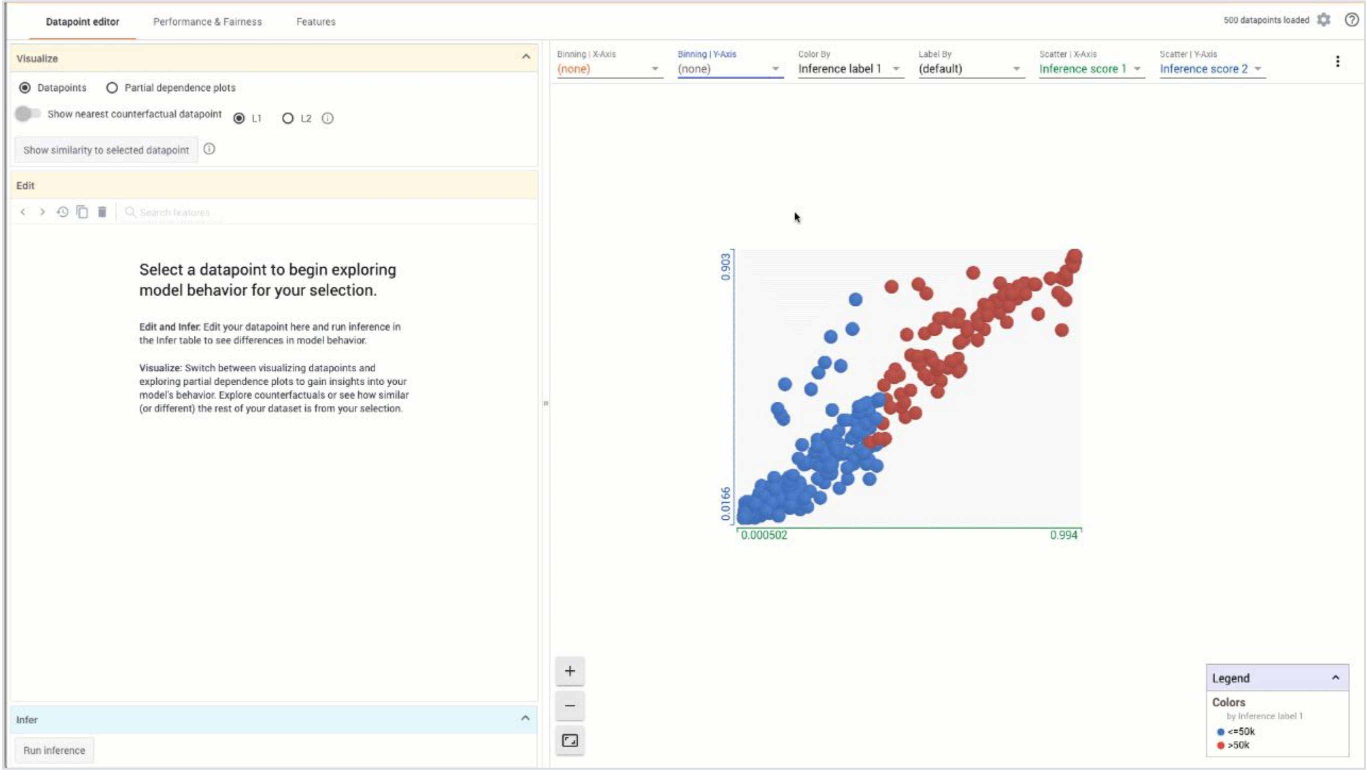
Compare performances of multiple models on the same simulation simultaneously.



Above: For classification problems, our counterfactual finding feature can identify the most similar datapoint (to a selection) in the loaded data that was classified differently by the model. For any dataset, L1 & L2 distances are available as inbuilt similarity metrics. However, users can specify custom metrics when invoking the tool.

Support many workflows without coding

Create custom visualizations using dataset features and model scores.



Above: Users can bin, scatter, color and label by any feature in the loaded dataset. This is useful for exploring the dataset and model results, as well as identifying biases in and suboptimal performance on specific slices of the dataset.

User-focused customizations

Ways to specify custom...

Models
Data
Distances
(eg. Similarity metrics)
Attributions
(eg. TCAV)
...

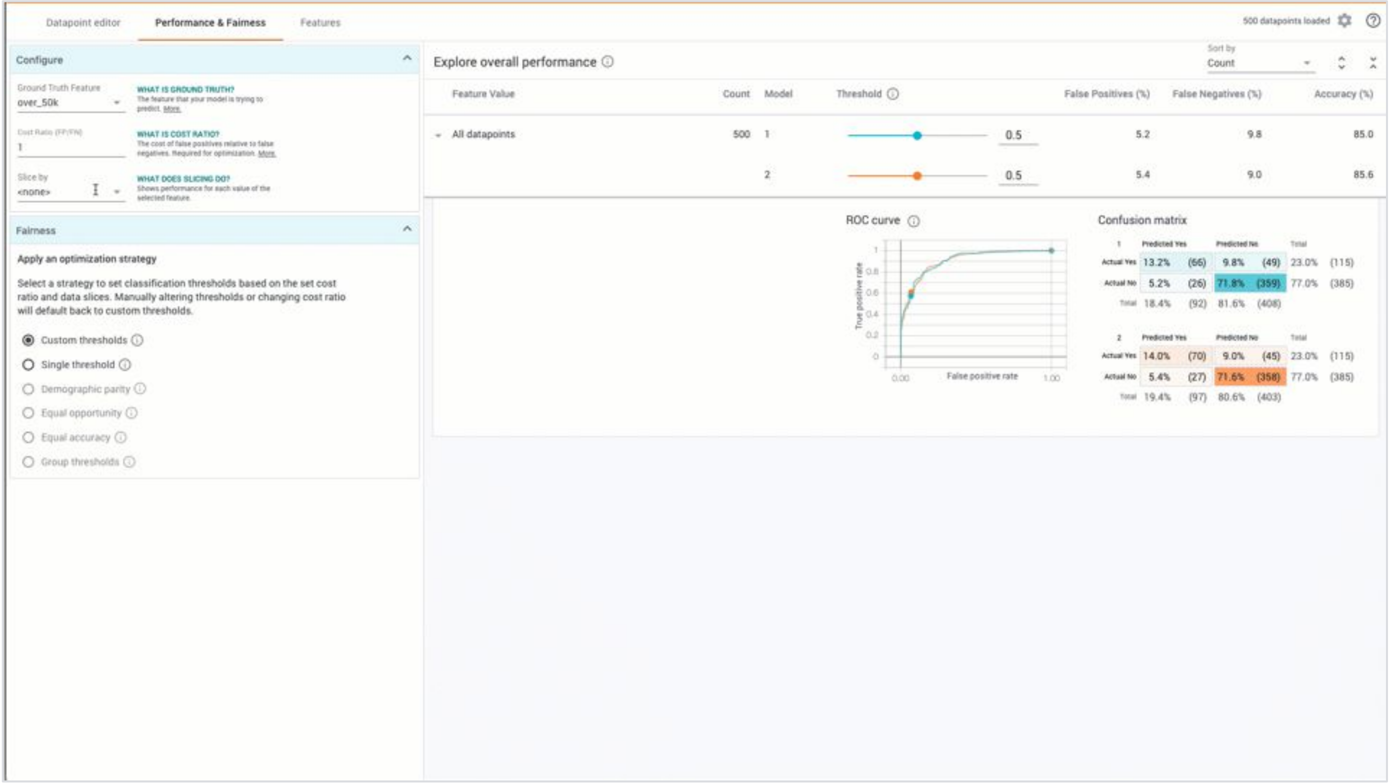
```
# This function extracts 'image/encoded' field, which is a reserved key for the
# feature that contains encoded image byte list. We read this feature into
# BytesIO and decode it back to an image using PIL.
# The model expects an array of images that are floats in range 0.0 to 1.0 and
# outputs a numpy array of (n_samples, n_labels)
def custom_predict(examples_to_infer):
  def load_byte_img(im_bytes):
    buf = BytesIO(im_bytes)
    return np.array(Image.open(buf), dtype=np.float64) / 255.

  ims = [load_byte_img(ex.features.feature['image/encoded'].bytes_list.value[0])
         for ex in examples_to_infer]
  preds = model1.predict(np.array(ims))
  return preds
```

Above: Example of a custom predict function in colaboratory

Visualizes
model performance

Allow user-defined intersectional analysis

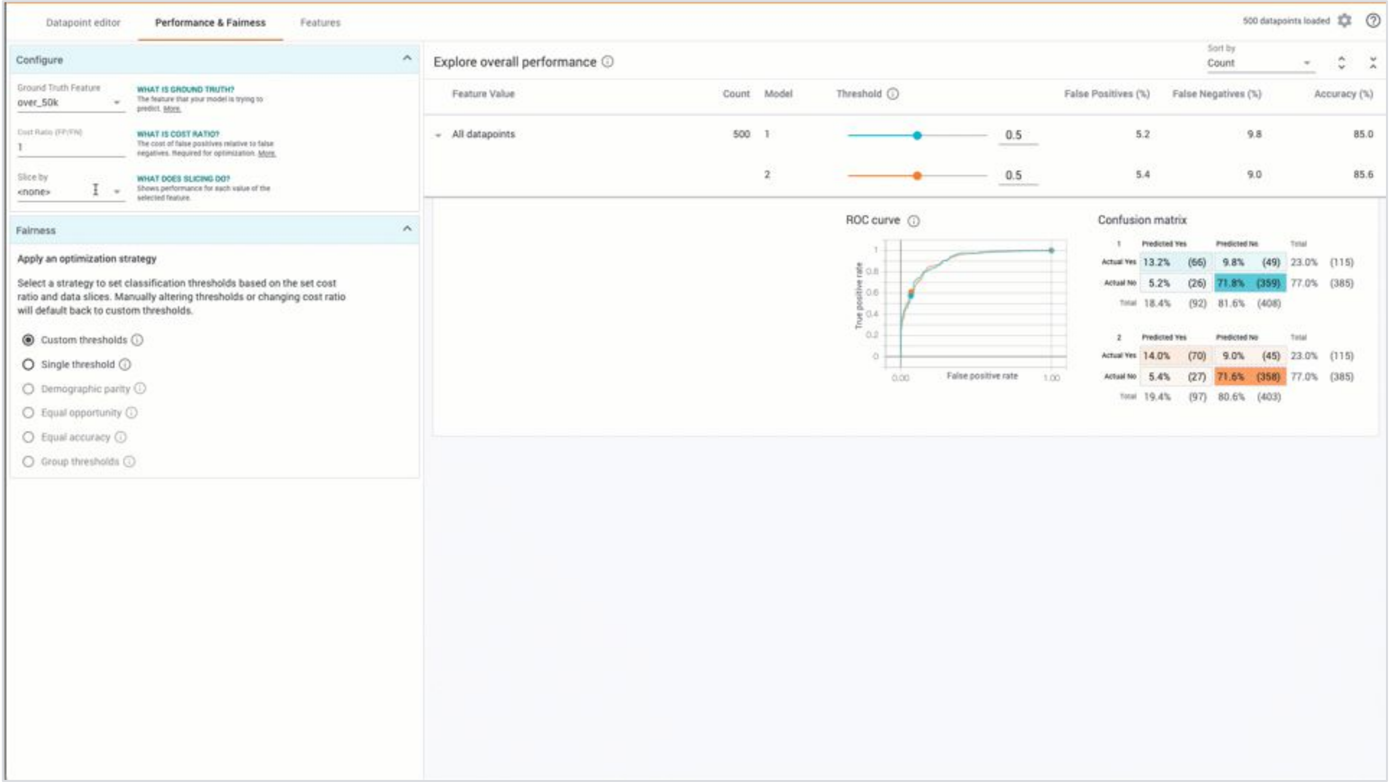
Evaluate performance on sub-groups of data rooted in feature values.



Above: Exploring model performance on different 'slices' of data given the values of specific features.

Explore ML Fairness optimizations

Translating fairness research into visual tooling.



Above: Exploring model performance on different ‘slices’ of data given the values of specific features.

Open-Source Tool

https://whatif-tool.dev

pip install witwidget



Q: What is **feature attribution?**

A: The amount each feature in a model contributes to the model's prediction.

For example

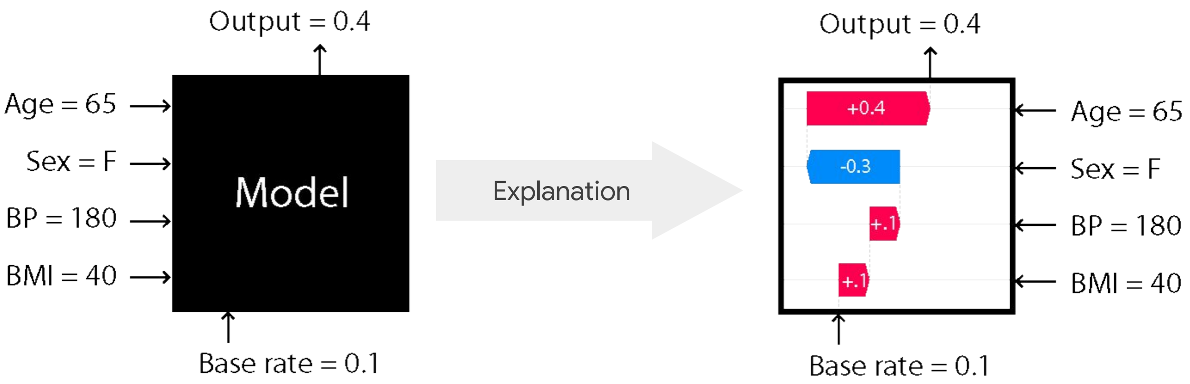
A model predicts you are 80% (0.8) likely to be approved for a loan.

Feature attributions:

Age	**0.3**
Income	**0.6**
Credit Score	**-0.1**

Q: What is **SHAP?**

A: An open source framework for inspecting
any machine learning model through feature attributions.



Q: **How does SHAP work?**

What does SHAP return?

SHAP assigns importance values to each feature indicating **the effect that feature had** on the model prediction.

How does SHAP calculate this?

SHAP approximates **the effect of removing a feature** from the model.

- Returns instance-level feature attributions along with global model-level feature importance.
- Works on image, text, and tabular models built with many different ML frameworks (TF, Scikit Learn, XGB, PyTorch),

--

Learn more: papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

#SHAP github.com/slundberg/shap

Q: How does SHAP work?

$$\phi_i(N, v) = \frac{1}{N!} \sum_{S \subseteq N \setminus \{i\}} |S|!(|N| - |S| - 1)! \Big[v(S \cup \{i\}) - v(S) \Big]$$

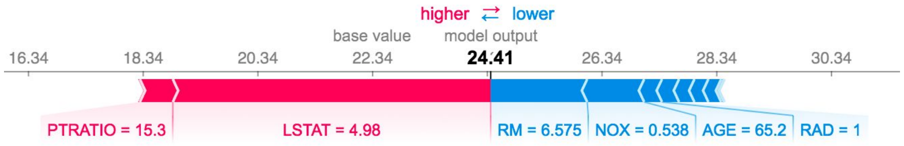
Weighted average of the marginal contribution for an agent.

How much does adding or removing a single feature affect the prediction?

--
Learn more: papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

Q: How does SHAP work?

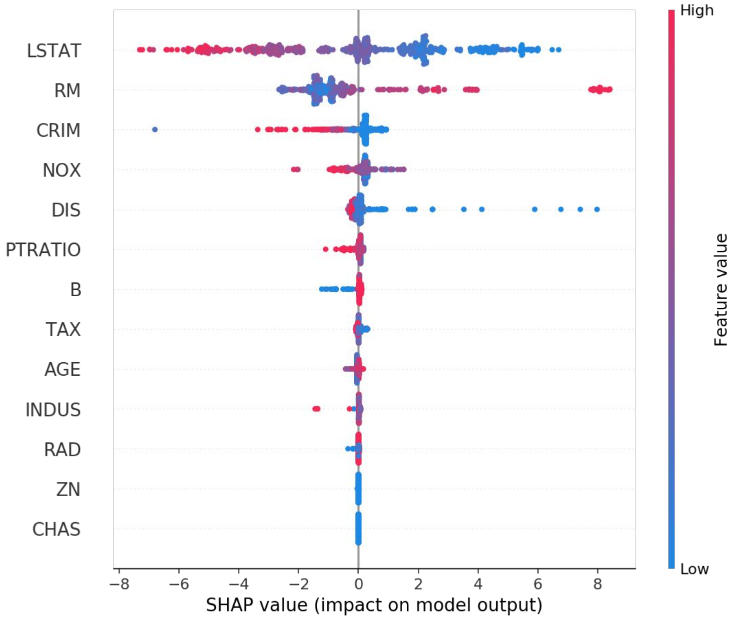
Instance-level attributions



Shows how each feature changes the model output from the baseline.

Red features pushed the prediction up from the baseline, blue features pushed it down.

Model-level attributions

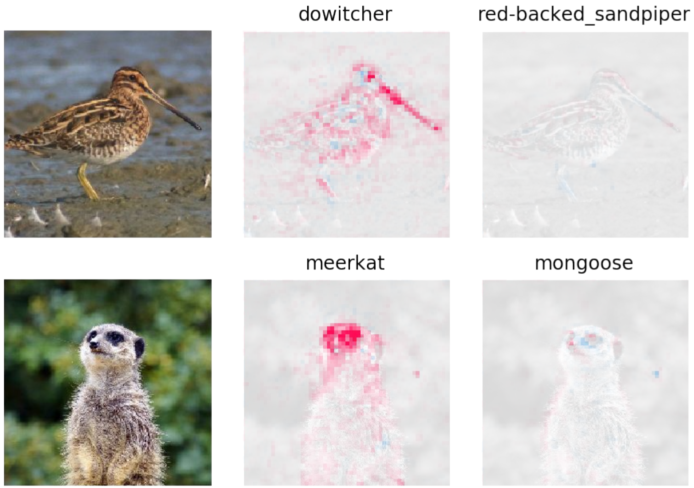


--

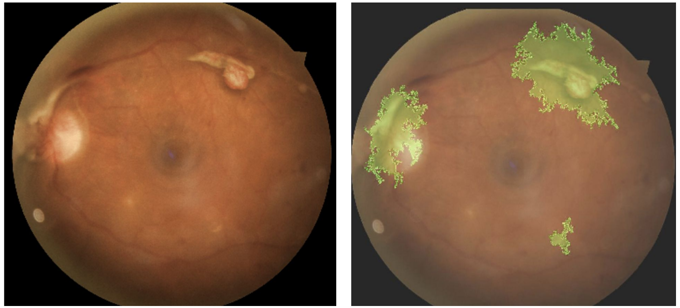
Learn more: [papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf]

Q: How might we interpret **image model attributions?**

Animal Species Classification Model



Diabetic Retinopathy (non-SHAP)



Getting started with SHAP

```
import shap

# Create the explainer
explainer = shap.DeepExplainer(model, train_data.values[:200])

# Get attribution values
shap_values = explainer.shap_values(train_data.values[:5])
```

Colab Notebook Exercise

Find link at:

https://pair-code.github.io/what-if-tool/fat2020.html

Caveats

Many approaches to interpreting models

Explainability is an emerging field with lots of ongoing research.

We've only shown a few methods.

Other techniques include: Integrated Gradients, LIME, SmoothGrad, etc...

Attribution techniques can be unreliable (see The (Un) reliability of saliency methods and related papers).

"ML fairness" doesn't solve societal issues

Making a model more fair has no effect on issues that may have caused creation of a problematic dataset.

Fairer models can still be used to treat people unfairly.

The world is not static - model decisions affect future situations. See the ML Fairness Gym project.

Discussion

What did we discover?

Any interesting patterns in model behavior?

What were the performance disparities between groups?

What features had the largest effect on predictions?

What ways did you use the tool to find insights?

How does this speak to the larger issues with this data/task?

Did anyone train a new model?

What differences in performance occurred?

What differences in attributions occurred?

How does this speak to the larger issues with this data/task?

Your feedback is important to us!
Scan the code below:

Thank You

More What-If Tool	**whatif-tool.dev**
People + AI Research	**ai.google.com/pair**
Get in touch	**groups.google.com/g/what-if-tool**
This feedback form	**forms.gle/ugkNHkVqJspK7v9Q9**