FACT-AI 2020

Week 2, Lecture 2: Transparency

Ana Lucic
University of Amsterdam

January 14, 2020

Overview

1. Motivation
2. Global Surrogate Explanation Models
3. Local Surrogate Explanation Models
   - LIME: Ribeiro et al. (2016)
   - SHAP: Lundberg and Lee (2017)
4. Counterfactual Examples
   - Laugel et al. (2019b)
   - Wachter et al. (2017)
   - Lucic et al. (2019)
5. Integrated Gradients
   - Sundararajan et al. (2017)
6. Additional Explanation Methods
7. Conclusion

Motivation

Figure: Husky or wolf?

Source: Ribeiro et al. (2016)

Motivation

Figure: Husky or wolf?

Prediction: Wolf
But can this prediction be *trusted*?

Source: Ribeiro et al. (2016)

Motivation

Figure: Left: original image, Right: explanation

The explanation shows us that the model is focusing on the (snowy) background, and not on the animal itself.

Source: Ribeiro et al. (2016)

Motivation

Figure: Left: original image, Right: explanation

The explanation shows us that the model is focusing on the (snowy) background, and not on the animal itself.
Correct answer: Husky!

Source: Ribeiro et al. (2016)

Global Surrogates

How can explanations for 'black-box' models be generated? A naive solution:

- Given a 'black-box' model f, train an interpretable model g using the predictions of f as the ground-truth for g.

Global Surrogates

How can explanations for 'black-box' models be generated? A naive solution:

- Given a 'black-box' model f, train an interpretable model g using the predictions of f as the ground-truth for g.
- Obtain an explanation through the interpretable model g (e.g., coefficients of linear model).

Global Surrogates

How can explanations for 'black-box' models be generated? A naive solution:

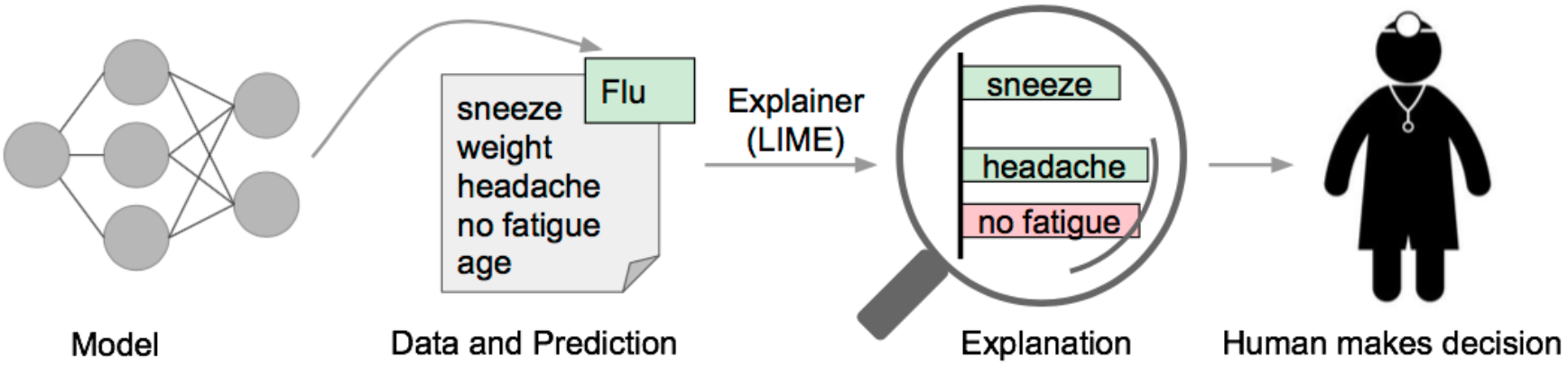
- Given a 'black-box' model f, train an interpretable model g using the predictions of f as the ground-truth for g.
- Obtain an explanation through the interpretable model g (e.g., coefficients of linear model).
- Evaluate g in terms of fidelity: the proportion of predictions from g that match the predictions of f.

Global Surrogates

How can explanations for 'black-box' models be generated? A naive solution:

- Given a 'black-box' model f, train an interpretable model g using the predictions of f as the ground-truth for g.
- Obtain an explanation through the interpretable model g (e.g., coefficients of linear model).
- Evaluate g in terms of fidelity: the proportion of predictions from g that match the predictions of f.
- However, as fidelity increases, added utility of using f instead of g decreases.

LIME: Local Interpretable Model Explainer



- Explains individual predictions by approximating them locally with a linear model.

- Given a test instance x, training set X, and complex model f, train a simple model g on a subset of X that is close to x.

Source: Ribeiro et al. (2016)

LIME: Local Interpretable Model Explainer

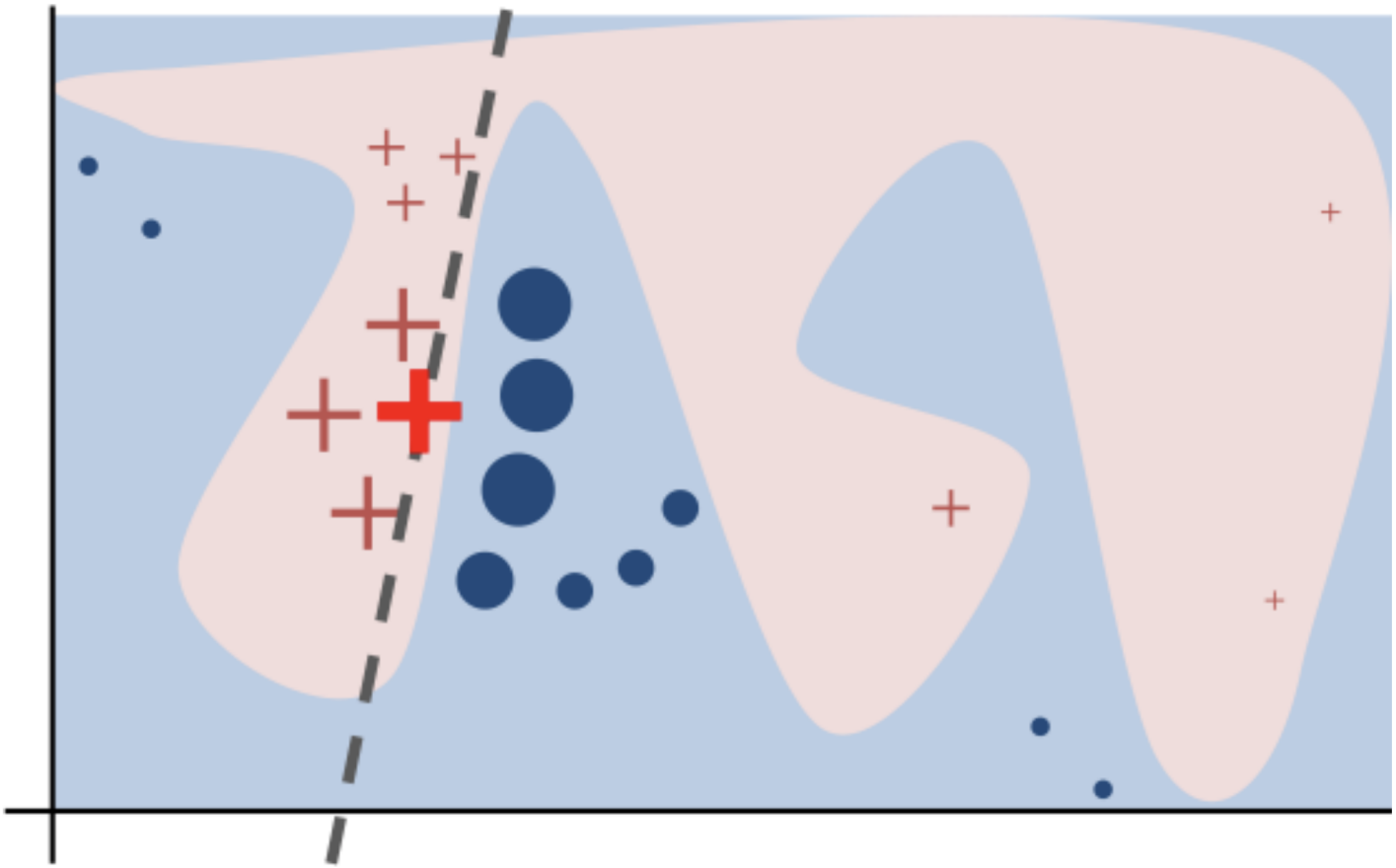


Figure: LIME explanation for the large red plus: fit a linear model mostly based on points near the original point, weighted by how close they are to the original point.

Source: Ribeiro et al. (2016)

LIME: Local Interpretable Model Explainer

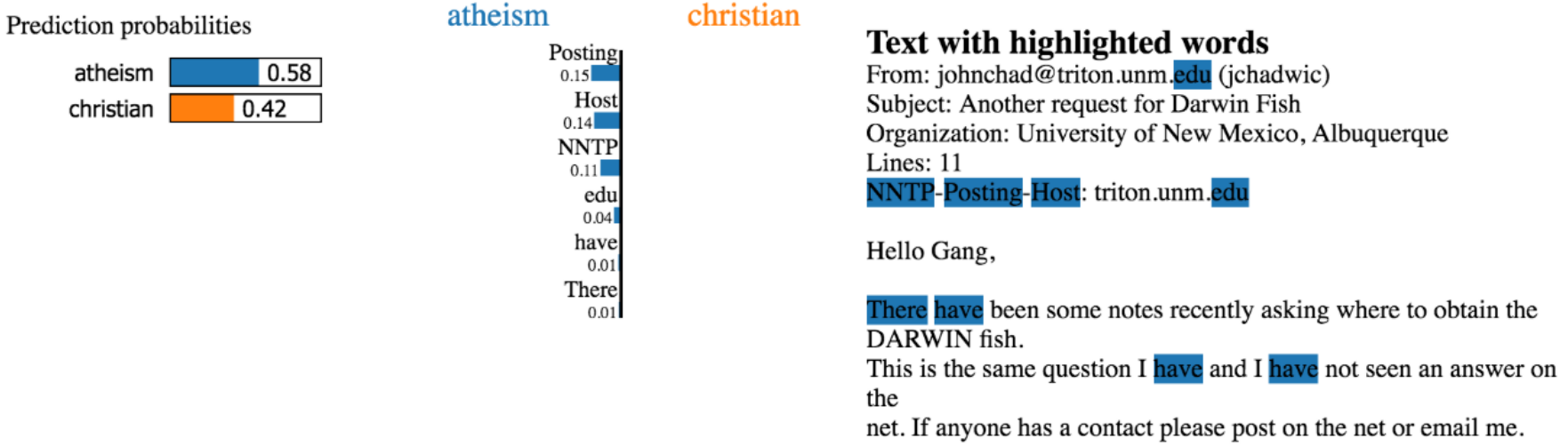


Figure: LIME explanation for a textual data: a binary classification task predicting whether an email is about Atheism or Christianity.

Source:
https://github.com/marcotcr/lime

LIME: Local Interpretable Model Explainer

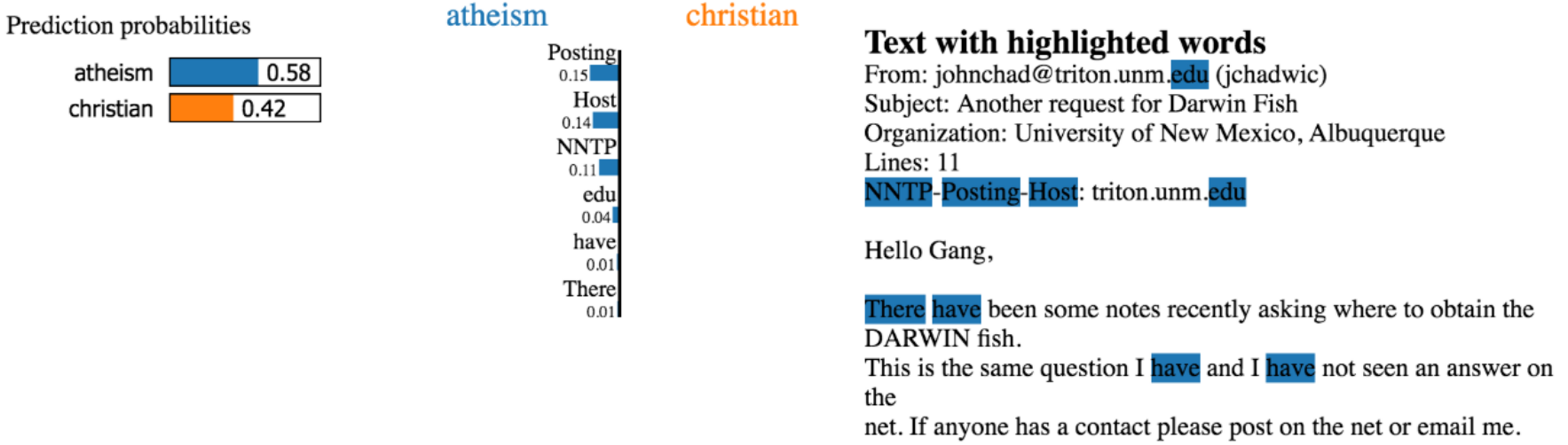
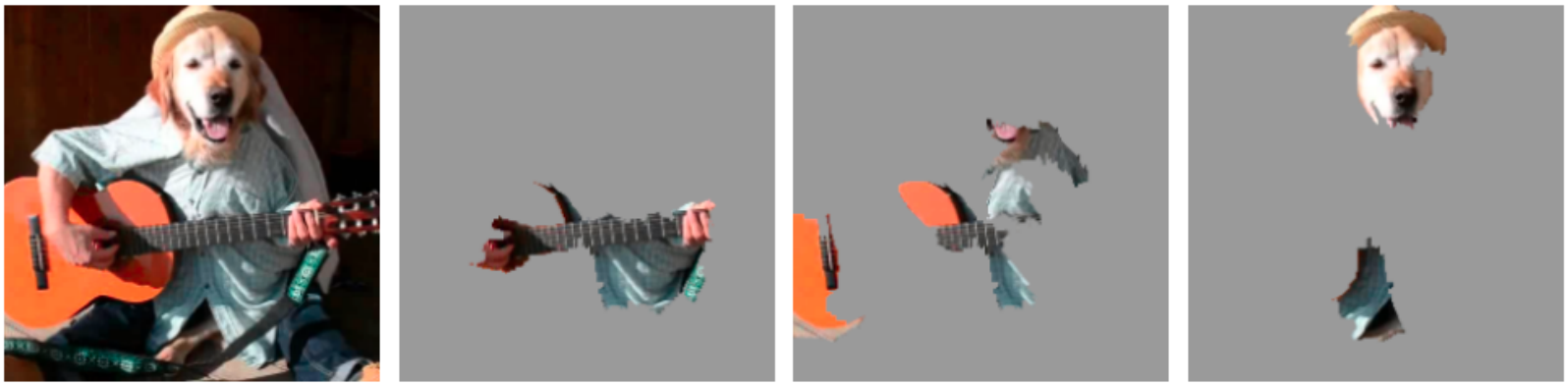


Figure: LIME explanation for a textual data: a binary classification task predicting whether an email is about Atheism or Christianity.

Explanations can help us make sure our models are *right for the right reasons*.

Source:
https://github.com/marcotcr/lime

LIME: Local Interpretable Model Explainer



(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure: LIME explanation for an image classification prediction. Top three predicted classes are "Electric guitar". "Acoustic guitar", and "Labrador".

Source: Ribeiro et al. (2016)

LIME: Local Interpretable Model Explainer

How does LIME fit into the taxonomy by Guidotti et al. (2018b)?

- **Problem type:**

LIME: Local Interpretable Model Explainer

How does LIME fit into the taxonomy by Guidotti et al. (2018b)?

- **Problem type:** Outcome (local) explanations
- **Explanator:**

LIME: Local Interpretable Model Explainer

How does LIME fit into the taxonomy by Guidotti et al. (2018b)?

- **Problem type:** Outcome (local) explanations
- **Explanator:** Feature importances
- **Model type:**

LIME: Local Interpretable Model Explainer

How does LIME fit into the taxonomy by Guidotti et al. (2018b)?

- **Problem type:** Outcome (local) explanations
- **Explanator:** Feature importances
- **Model type:** Model-agnostic
- **Data type:**

LIME: Local Interpretable Model Explainer

How does LIME fit into the taxonomy by Guidotti et al. (2018b)?

- **Problem type:** Outcome (local) explanations
- **Explanator:** Feature importances
- **Model type:** Model-agnostic
- **Data type:** Text, Image, Tabular

SHAP: Shapley Additive Values

- SHAP is another explanation method that generates feature importances for individual predictions.
- Can be aggregated to produce global explanations for an entire dataset.
- Strong theoretical guarantees (see Section 3 of Lundberg and Lee (2017)).
- Widely regarded as the current SOTA for feature importance explanation methods.

SHAP: Shapley Additive Values

SHAP is inspired by work in game theory about attributing surplus in a cooperative game:

Figure: Total surplus when all three players cooperate.

Source: Lo (2014)

SHAP: Shapley Additive Values

SHAP is inspired by work in game theory about attributing surplus in a cooperative game:

Figure: Total surplus when all three players cooperate.

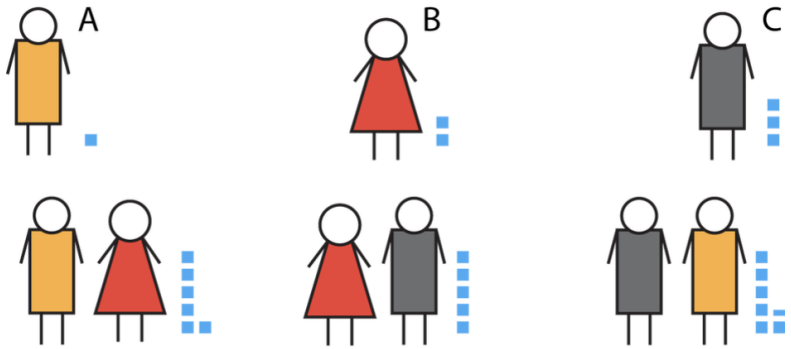


Figure: Surplus for each possible subset of players

Note that players' contributions to the total surplus are **not** independent of one another!

Source: Lo (2014)

SHAP: Shapley Additive Values

- Shapley values tell us how much of the total surplus each player is *responsible for*.
- Lipovetsky and Conklin (2001) applied this idea to regression outputs: **how much does each feature contribute to the overall prediction?**
- Lundberg and Lee (2017) propose efficient ways to approximate Shapley values specifically for linear models, deep models and tree-based models.

SHAP: Shapley Additive Values

SHAP provides both coarse and granular feature importance visualizations:

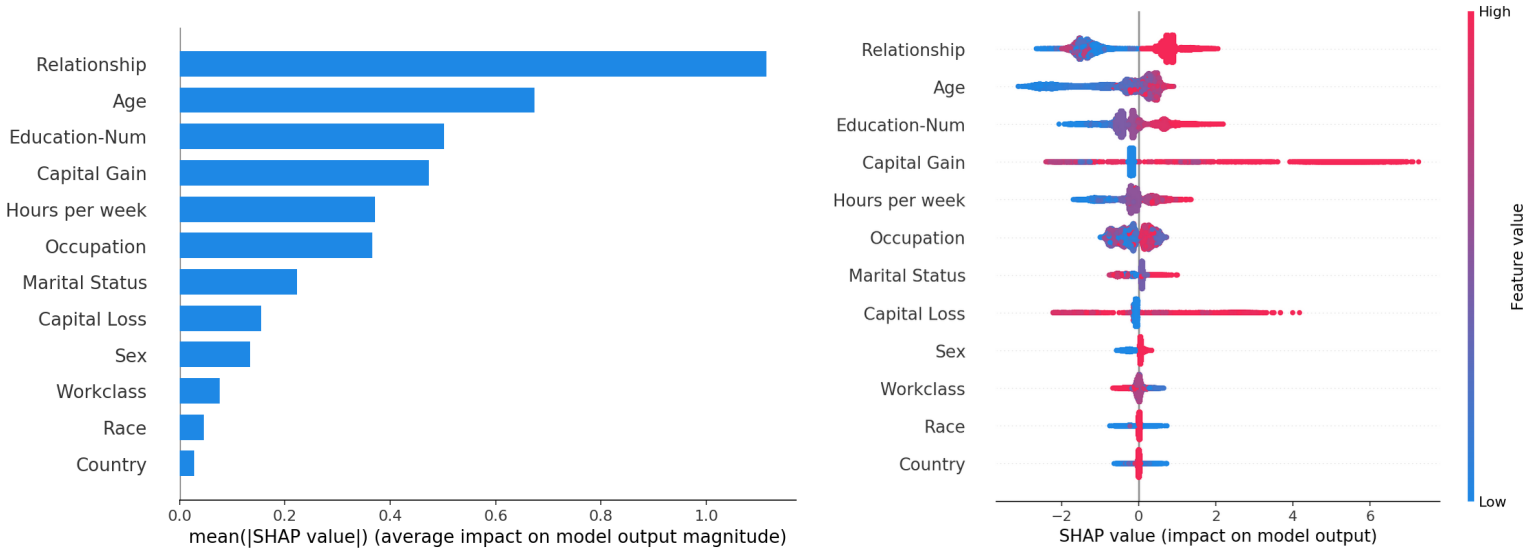


Figure: Left: Summary plot of SHAP values, Right: Density plot of SHAP values.

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

Counterfactual Examples

- Explanations should be actionable – given an explanation, a user should be able to understand what they need to change in order to change the outcome.
- Counterfactual explanations are based on finding counterfactual examples that are close to the original example but have a different prediction.

Counterfactual examples

Given a model f and an instance x, find x' such that $f(x) \neq f(x')$ and $d(x, x')$ is minimal, where d is some distance function.

Definition 1: Counterfactual example

The minimal perturbation required to change the predicted class of a given observation (i.e., given $x \in X$, the counterfactual example is x').

Definition 2: Counterfactual explanation

The difference between x and x'.

Counterfactual Explanations

Ideally, a counterfactual example x' should satisfy the following properties (Laugel et al., 2019b):

- **Proximity:** x' should be close to actual training examples that are in the same class.
- **Connectedness:** x' should not live in a part of the decision space where there do not exist training examples.
- **Stability:** two instances x_1 and x_2 are similar $\implies$ x_1' and x_2' are similar.

Counterfactual Explanations

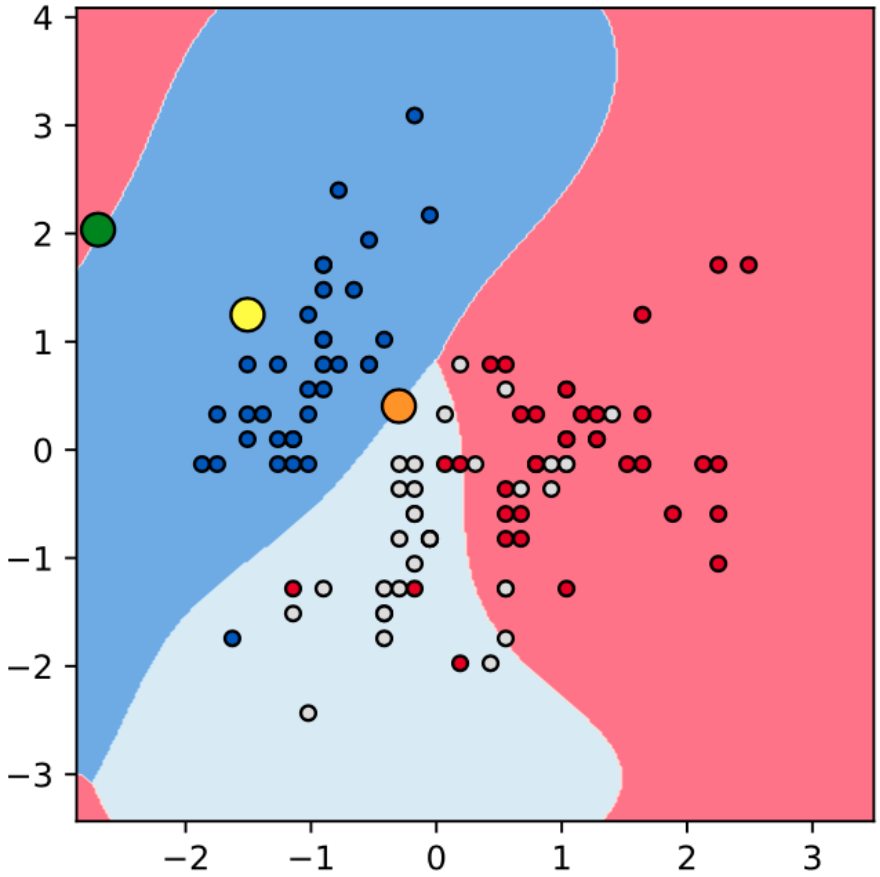


Figure: Decision boundaries for Iris dataset. Yellow dot is the original x. Green and orange are counterfactual examples.

Source: Laugel et al. (2019a)

Counterfactual Explanations

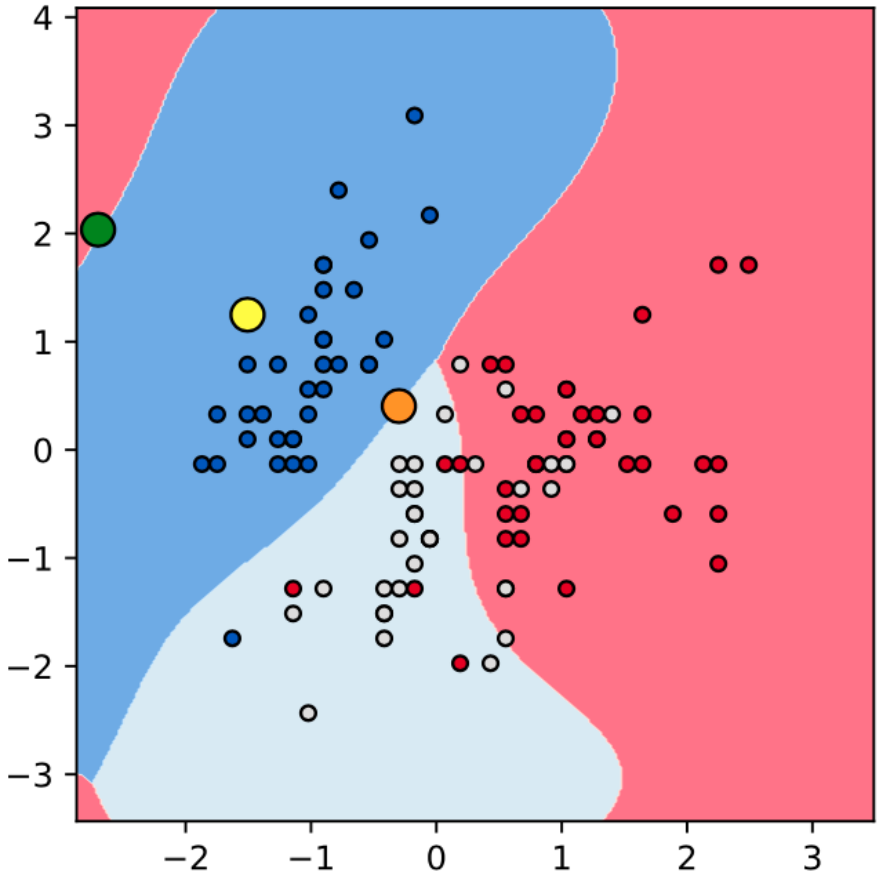


Figure: Decision boundaries for Iris dataset. Yellow dot is the original x. Green x' is not a good counterfactual example since it does not satisfy the connectedness property. Orange x' satisfies proximity and connectedness.

Source: Laugel et al. (2019a)

Counterfactual Examples

Wachter et al. (2017) propose generating counterfactual examples by minimizing a loss function of the form:

$$\mathcal{L}_{total}(f, x, x') = \mathcal{L}_{prediction}(f, x, x') + \mathcal{L}_{distance}(x, x')$$

where:

- $\mathcal{L}_{prediction}$ = any differentiable prediction loss function (e.g. hinge loss, mean squared error)
- $\mathcal{L}_{distance}$ = any differentiable distance function (e.g. Euclidean distance, Cosine distance)

Counterfactual Examples

Different distance functions can produce different counterfactual examples (Lucic et al., 2019).

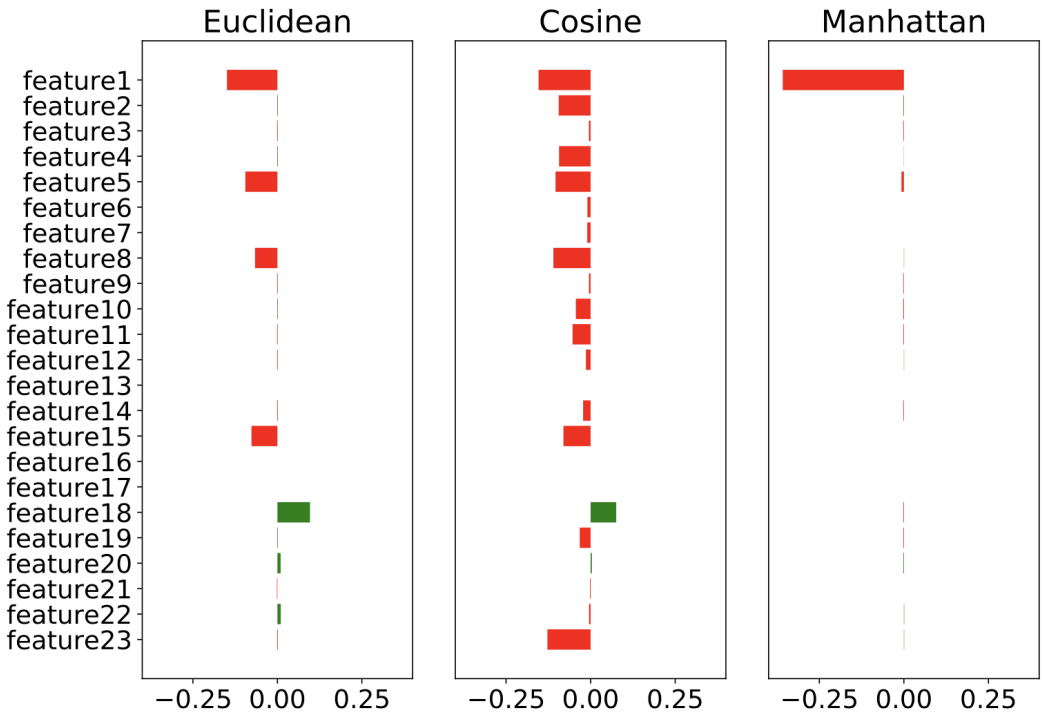


Figure: Euclidean, Cosine and Manhattan counterfactual explanations for the same input instance. Red $\Longrightarrow$ decrease feature value, green $\Longrightarrow$ increase feature value.

Source: Lucic et al. (2019)

Integrated Gradients

Given an image, determine the pixels that contributed to the prediction:

Figure: Image predicted as “fireboat”

Source: https://github.com/ankurtaly/Integrated-Gradients

Integrated Gradients

Interpolate a series of images between a baseline (all black) image and the original image, varying in intensity.

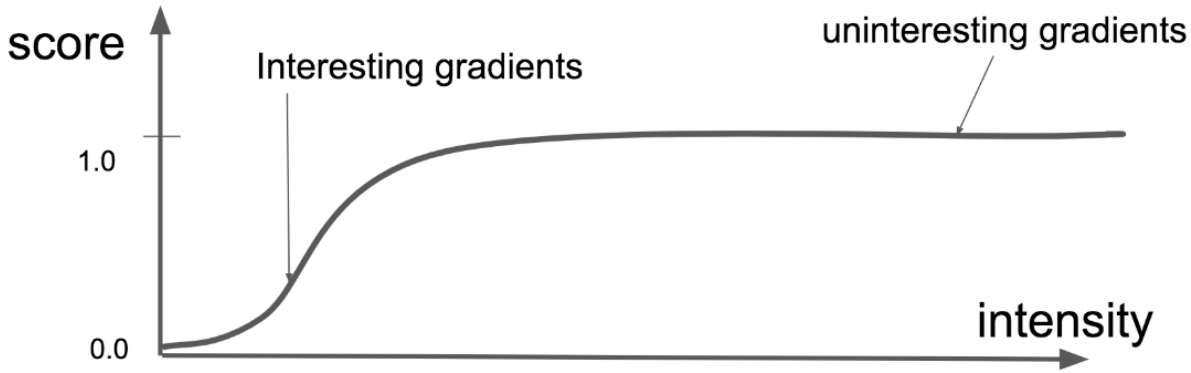


Source: `https://github.com/ankurtaly/Integrated-Gradients`

Integrated Gradients

Interpolate a series of images between a baseline (all black) image and the original image, varying in intensity.

Plot the softmax score for the predicted category vs. the intensity



Source: https://github.com/ankurtaly/Integrated-Gradients

Integrated Gradients

Take the gradient of the final output w.r.t. the interpolated images:

Figure: Scaled images: from baseline to original

Figure: Scaled gradients of output w.r.t. images

Source: `https://github.com/ankurtaly/Integrated-Gradients`

Integrated Gradients

Some more examples of Integrated Gradients explanations:

Top label: starfish

Score: 0.999992

Top label: school bus

Score: 0.997033

Top label: mosque

Score: 0.999127

Source: Sundararajan et al. (2017)

Additional Explanation Methods

- **Prototypes:** Li et al. (2018)
- **Influential training points:** Koh and Liang (2017), Sharchilev et al. (2018)
- **Decision sets/rules:** Lakkaraju et al. (2019), Guidotti et al. (2018a)
- **Concepts:** Ghorbani et al. (2019)
- **Available Toolboxes:** Nori et al. (2019), Arya et al. (2019)

Conclusion

- Complex models should only be used for complex problems; complex models are unnecessary if a simple (and interpretable) model can solve the task at hand.
- Most existing methods for interpreting complex models involve interpreting individual predictions.
- There exist many, many more methods than those discussed here today, and there does not exist a single one-fits-all solution to transparency in AI $\implies$ **lots of room to contribute!**

GPUs for Assignment

If you need access to GPUs for your project, please fill in this form below by 12:00 today: `https://forms.gle/xiWaBzdsYSTWZgMb6`

References I

Arya, V., Bellamy, R. K. E., Chen, P.-Y., Dhurandhar, A., Hind, M., Hoffman, S. C., Houde, S., Liao, Q. V., Luss, R., Mojsilović, A., Mourad, S., Pedemonte, P., Raghavendra, R., Richards, J., Sattigeri, P., Shanmugam, K., Singh, M., Varshney, K. R., Wei, D., and Zhang, Y. (2019). One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques. *arXiv:1909.03012 [cs, stat]*. arXiv: 1909.03012.

Ghorbani, A., Wexler, J., Zou, J., and Kim, B. (2019). Towards Automatic Concept-based Explanations. *NeurIPS 2019*. arXiv: 1902.03129.

Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., and Giannotti, F. (2018a). Local Rule-Based Explanations of Black Box Decision Systems. *arXiv:1805.10820 [cs]*. arXiv: 1805.10820.

Guidotti, R., Monreale, A., Turini, F., Pedreschi, D., and Giannotti, F. (2018b). A Survey Of Methods For Explaining Black Box Models.

Koh, P. W. and Liang, P. (2017). Understanding Black-box Predictions via Influence Functions.

References II

Lakkaraju, H., Kamar, E., Caruana, R., and Leskovec, J. (2019). Faithful and Customizable Explanations of Black Box Models. *AAAI 2019*.

Laugel, T., Lesot, M.-J., Marsala, C., and Detyniecki, M. (2019a). Issues with post-hoc counterfactual explanations: a discussion. *arXiv:1906.04774 [cs, stat]*. arXiv: 1906.04774.

Laugel, T., Lesot, M.-J., Marsala, C., Renard, X., and Detyniecki, M. (2019b). The Dangers of Post-hoc Interpretability: Unjustified Counterfactual Explanations. *arXiv:1907.09294 [cs, stat]*. arXiv: 1907.09294.

Li, O., Liu, H., Chen, C., and Rudin, C. (2018). Deep Learning for Case-Based Reasoning through Prototypes: A Neural Network that Explains Its Predictions. *AAAI 2018*.

Lipovetsky, S. and Conklin, M. (2001). Analysis of Regression in Game Theory Approach.

Lo, R. (2014). Group project - how much did i contribute?

References III

Lucic, A., Oosterhuis, H., Haned, H., and de Rijke, M. (2019). Actionable Interpretability through Optimizable Counterfactual Explanations for Tree Ensembles. *arXiv:1911.12199 [cs]*.

Lundberg, S. M. and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *NeurIPS 2017*.

Nori, H., Jenkins, S., Koch, P., and Caruana, R. (2019). InterpretML: A Unified Framework for Machine Learning Interpretability. *arXiv:1909.09223 [cs, stat]*. arXiv: 1909.09223.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why Should I Trust You?" Explaining the Predictions of Any Classifier. *KDD 2016*.

Sharchilev, B., Ustinovsky, Y., Serdyukov, P., and de Rijke, M. (2018). Finding influential training samples for gradient boosted decision trees. In *ICML*, pages 4584–4592.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic Attribution for Deep Networks. *ICML 2017*.

References IV

Wachter, S., Mittelstadt, B., and Russell, C. (2017). Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *SSRN Electronic Journal*.