Four Machine Learning Techniques that Tackle Scale

(And Not Just By Increasing Accuracy)

Lindsey Lack

Gigamon Applied Threat Research (ATR)

99.9%

	Predict: benign	Predict: malicious
Actual: benign	998940	910
Actual: malicious	90	60

False discovery rate 0.938

	Predict: benign	Predict: malicious
Actual: benign	998860	990
Actual: malicious	10	140

False discovery rate 0.876

Gigamon

99.99%

Gigamon

	Predict: benign	Predict: malicious
Actual: benign	999759	91
Actual: malicious	9	141

False discovery rate 0.392

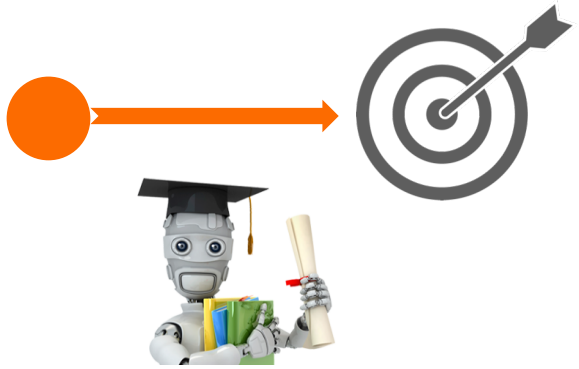
	Predict: benign	Predict: malicious
Actual: benign	999825	25
Actual: malicious	75	75

False discovery rate 0.250

Gigamon

Intended Audience

Cybersecurity-to-Data Science Continuum



Overview

Measurement

Explainability

Confidence

Architecture

Measurement

	Predict: benign	Predict: malicious
Actual: benign	999759	91
Actual: malicious	9	141

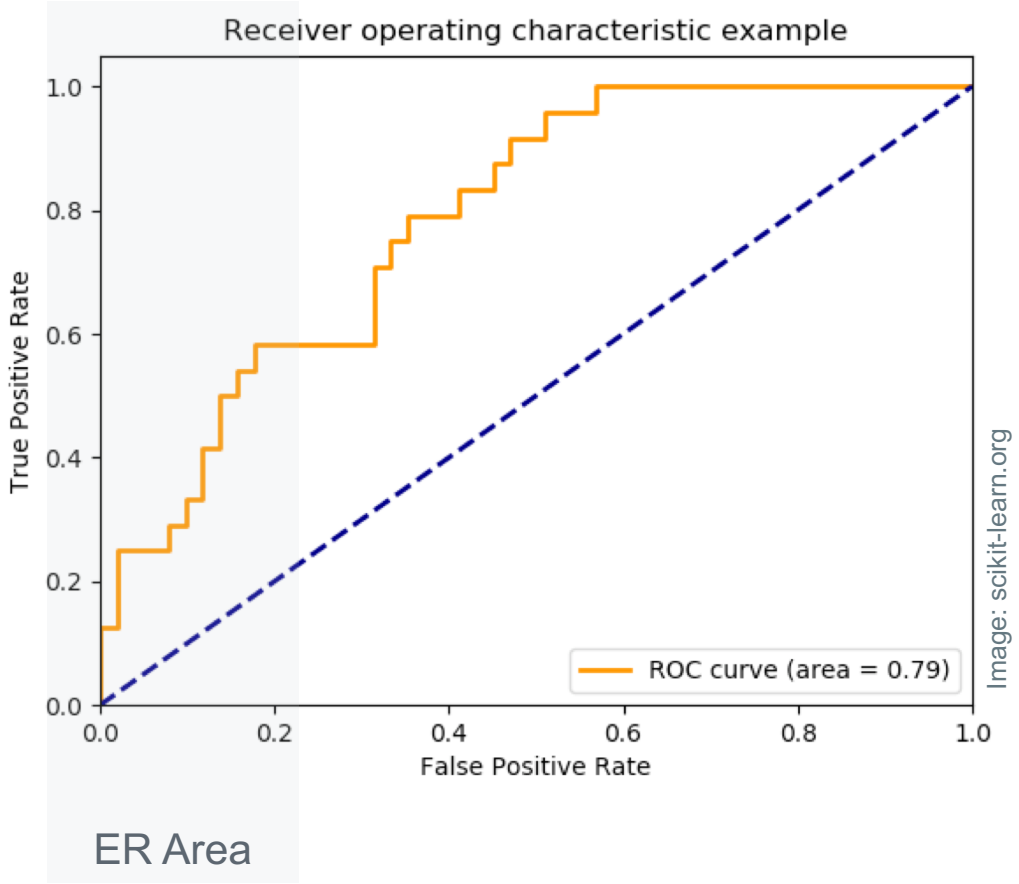
Precision 0.608
Recall 0.940

	Predict: benign	Predict: malicious
Actual: benign	999825	25
Actual: malicious	75	75

Precision 0.750
Recall 0.500

Gigamon

ROC Curve



ER Area

Cons:

- Requires fine-grained output (“proba”)
- AUC changes not intuitive

Pros:

- Visualize FP and FN tradeoffs
- Allows for focus on false positives
  - (Early Retrieval area)

Gigamon

ROC Curves with Highlighted FPR

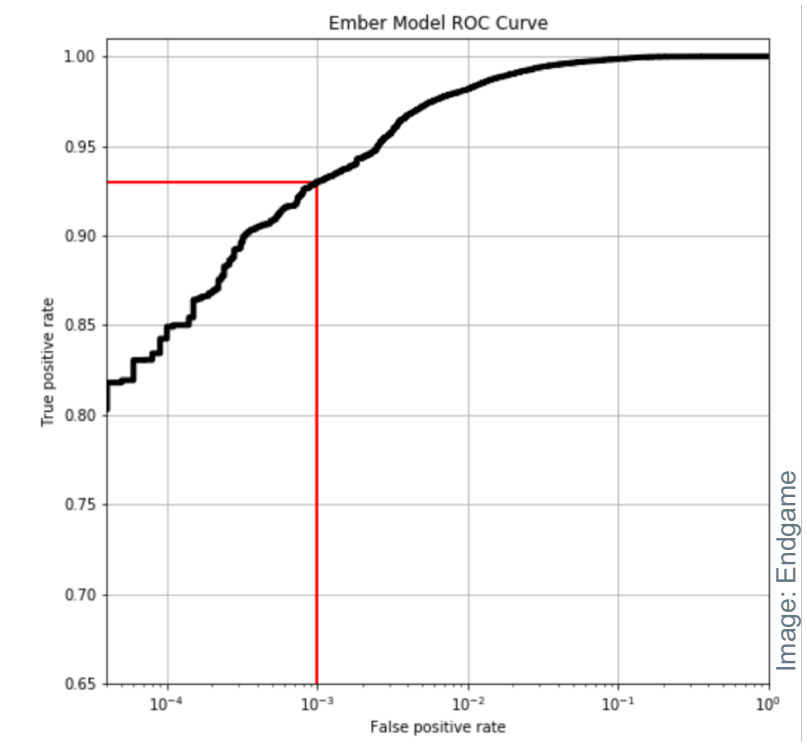
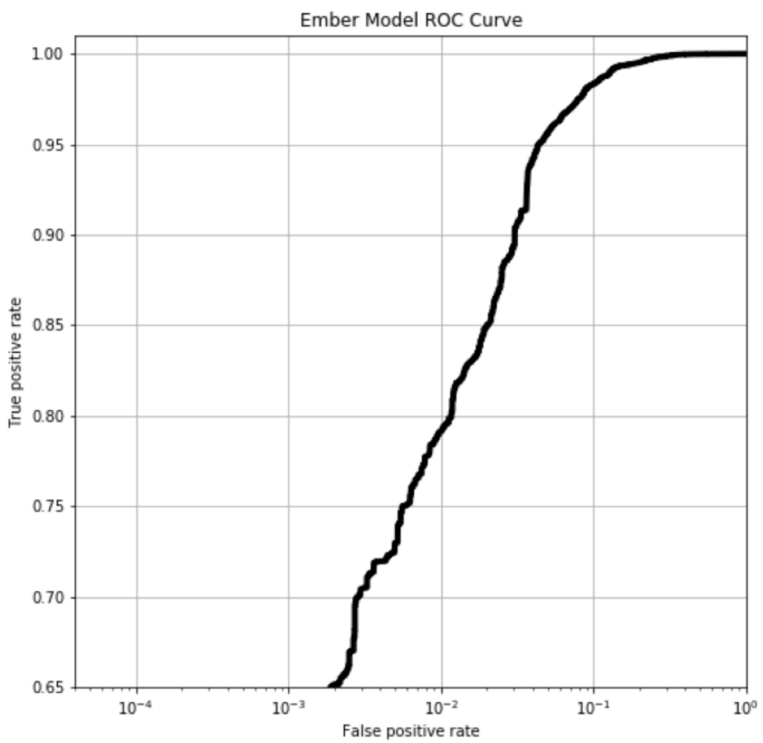


Image: Endgame



Full features

Reduced features

Measurement

Takeaway

Measure items in a way that is consistent with operational goals.

For security use cases, this may entail a focus on false positives.

Explainability

AKA Interpretability

The black box issue is in the news

Image: Information Age

Image: TNW

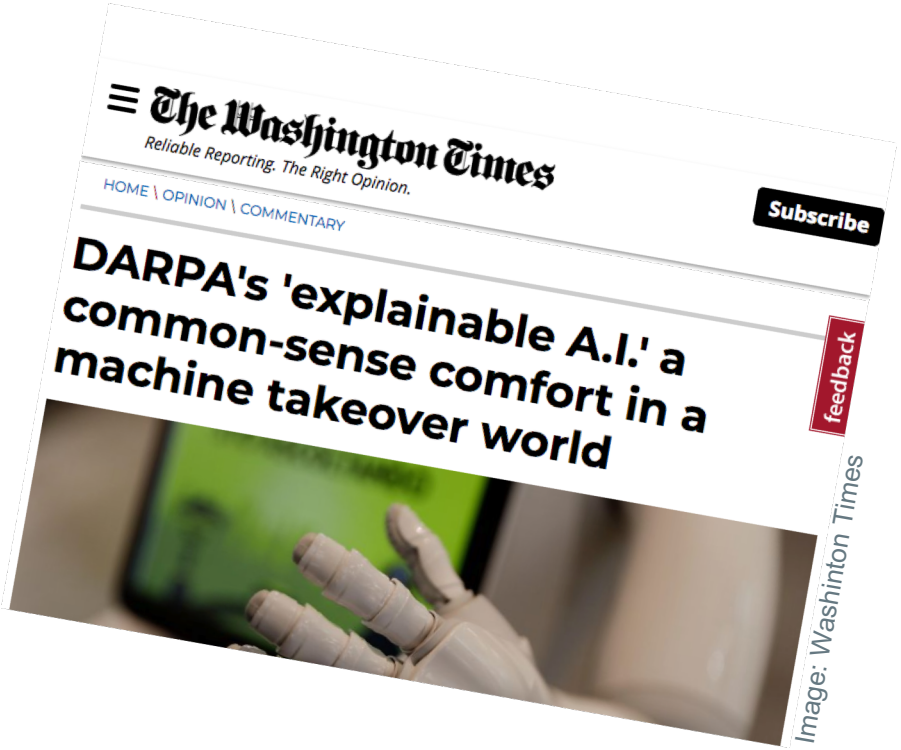


Image: Washinton Times

Image: DARPA

Model Explanations

The Basics

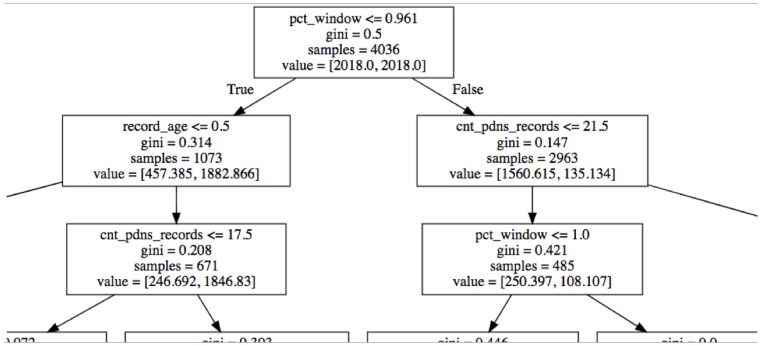
Built-In Feature Ranking

- Valid for whole model only, does not inform for particular instance

> **`feature_importances_`** : ***array of shape = [n_features]***
>
> Return the feature importances (the higher, the more important the feature).

Train and Examine Simpler Model

- E.g. Decision tree



Model Explanations

Recent Purpose-built Tools

LIME

- 2016
- Local Interpretable Model-Agnostic Explanations

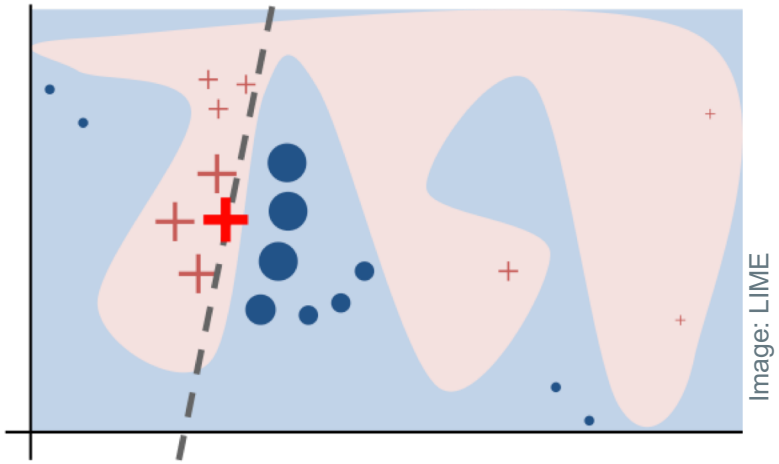


Image: LIME

SHAP

- 2017
- **SH**apley **A**dditive ex**P**lanation (SHAP) Values
- An extension of the Shapley values method
- Uses game theory and notion of fair ‘payouts’

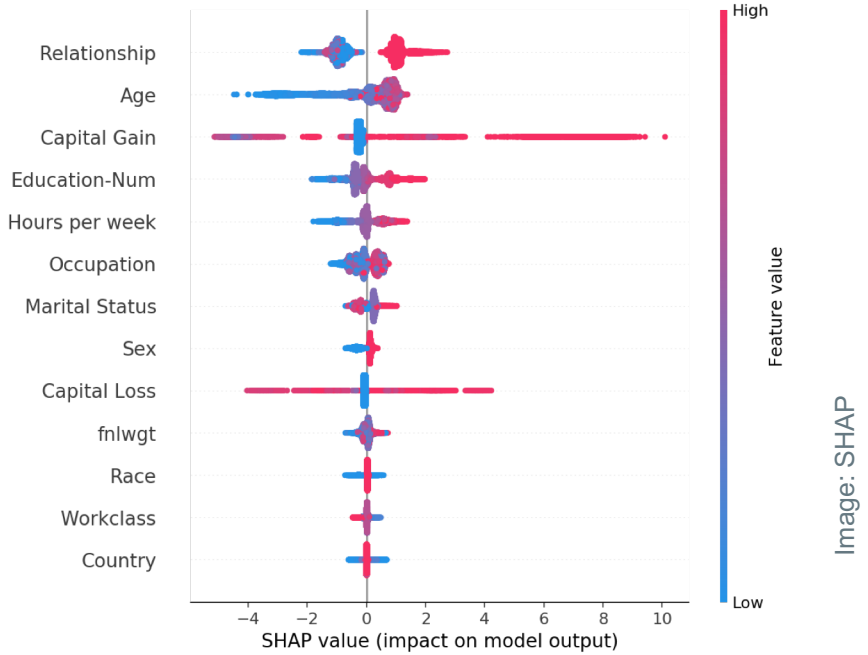


Image: SHAP

Model Explanations

Endgame SHAP analysis of Ember model

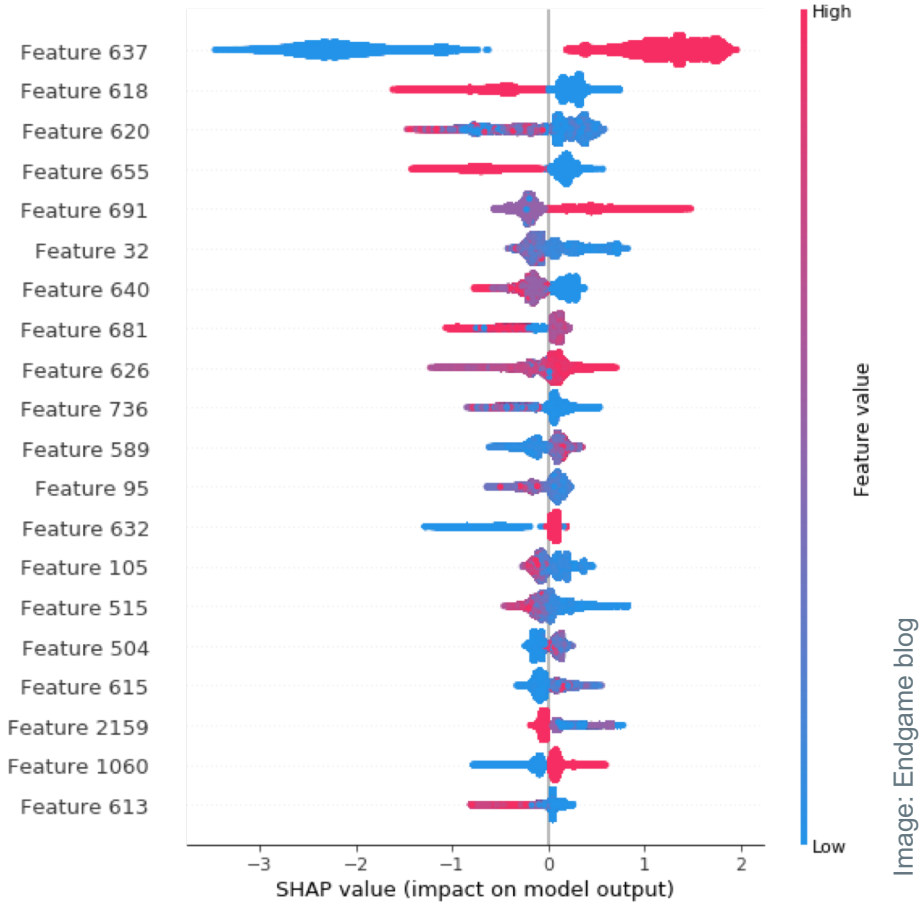
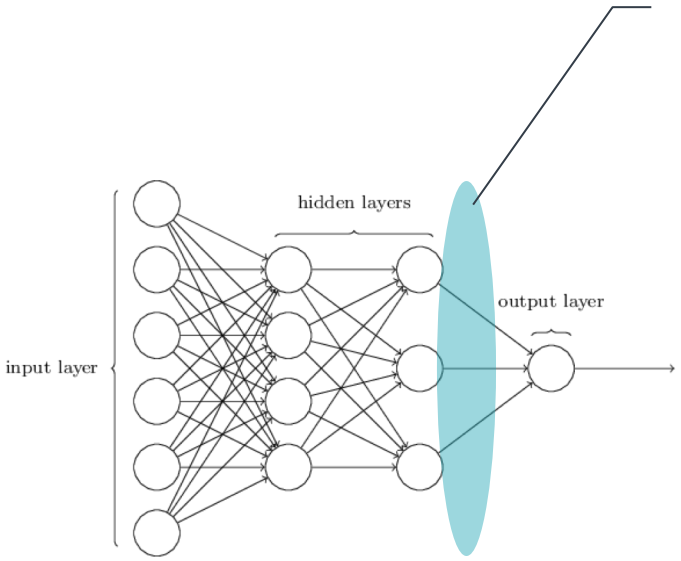


Image: Endgame blog

Image: Endgame blog

Model Explanations

Use of trained layers for effective Nearest Neighbor calculations

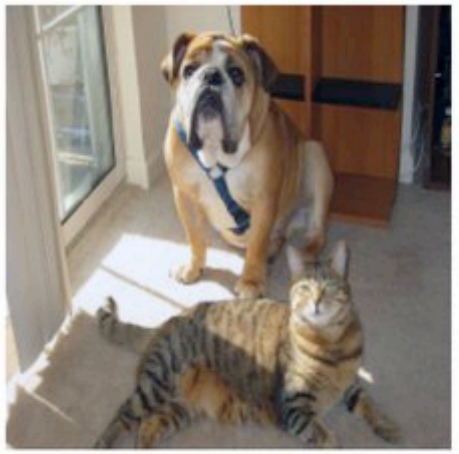


Use trained penultimate layer output as instance vector for distance calculation

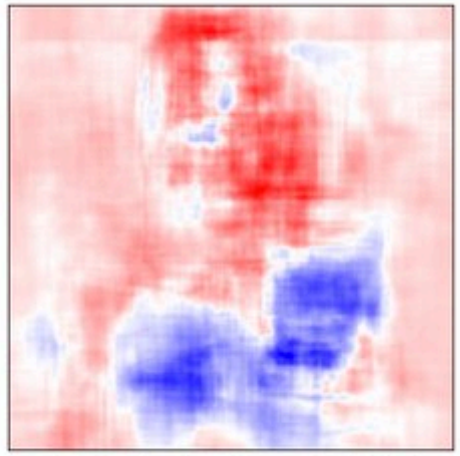
Malicious URI and Nearest Neighbors in Training Set	
Distance	URI
	www.0576pet.cn/view-13285-1.html
0.094	www.0817auto.cn/view-12858701.html
0.144	mythproductionhouse.com/pre-win-error-page-al…
0.158	www.tamizhtube.com/search/label/\xe0\xae\xb5\...

Model Explanations

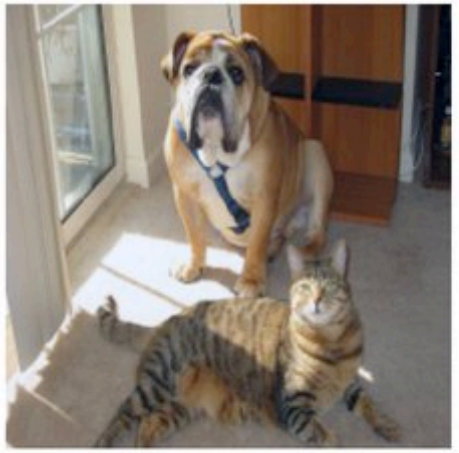
Images: Basic technique occlusion mapping



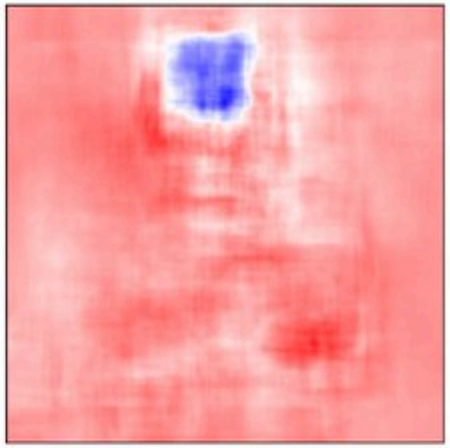
(a) Original Image



(e) Occlusion Map for 'Cat'



(f) Original Image

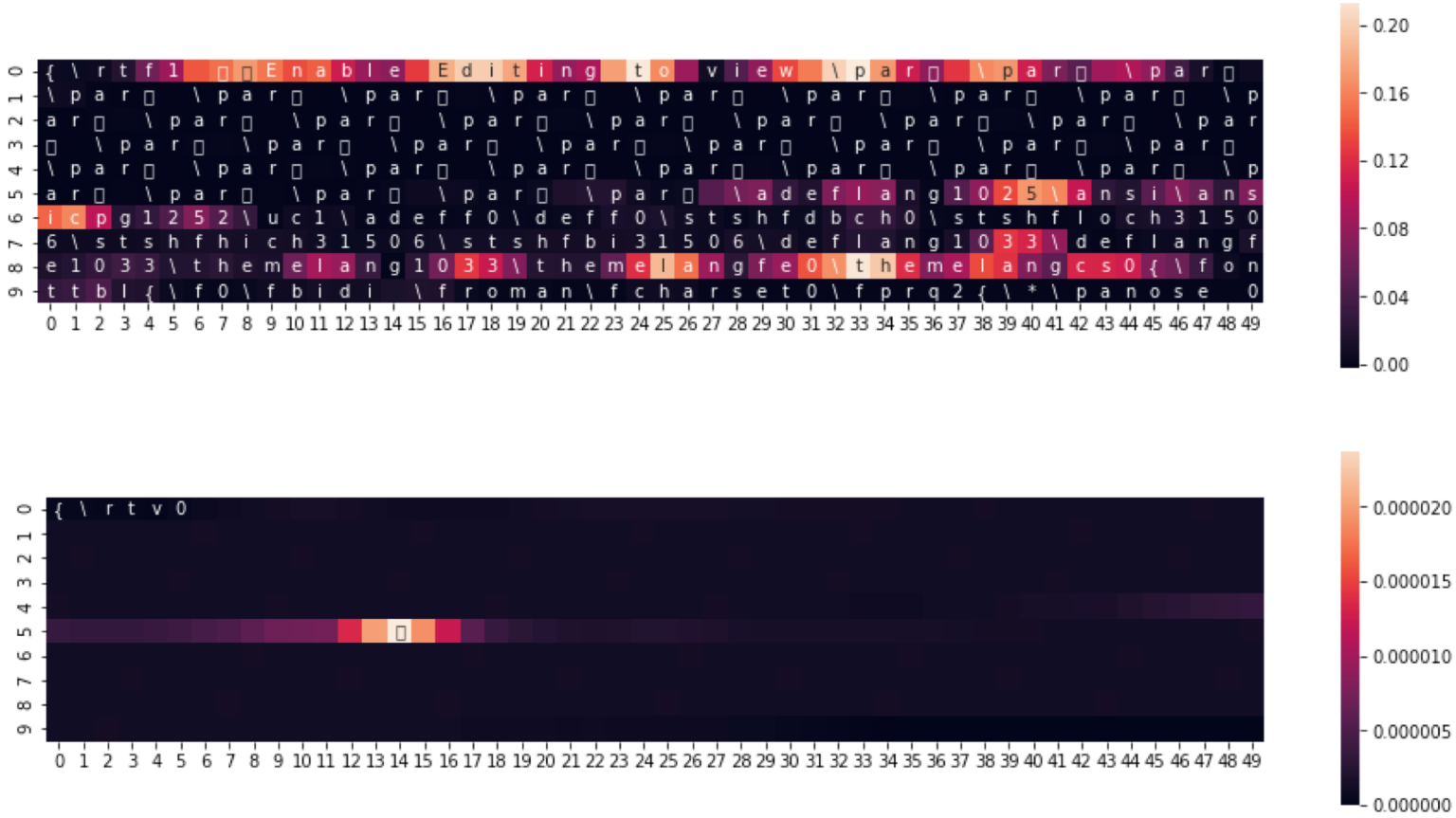


(j) Occlusion Map for 'Dog'

Image: Selvaraju et al

Model Explanations

Occlusion-based Analysis of Malicious RTF files



Credit: Zeiler et al, 2014

Explainability

Takeaway

Don’t settle for solitary outputs (label or single probability) from models. Provide model context or insight (many methods available) that allows an analyst to “scan” results.

Confidence

Model probability output
!=
confidence

Most entertaining example of confidence mismatch:

"Passing a Chicken through an MNIST Model" blog entry by Emilien Dupont

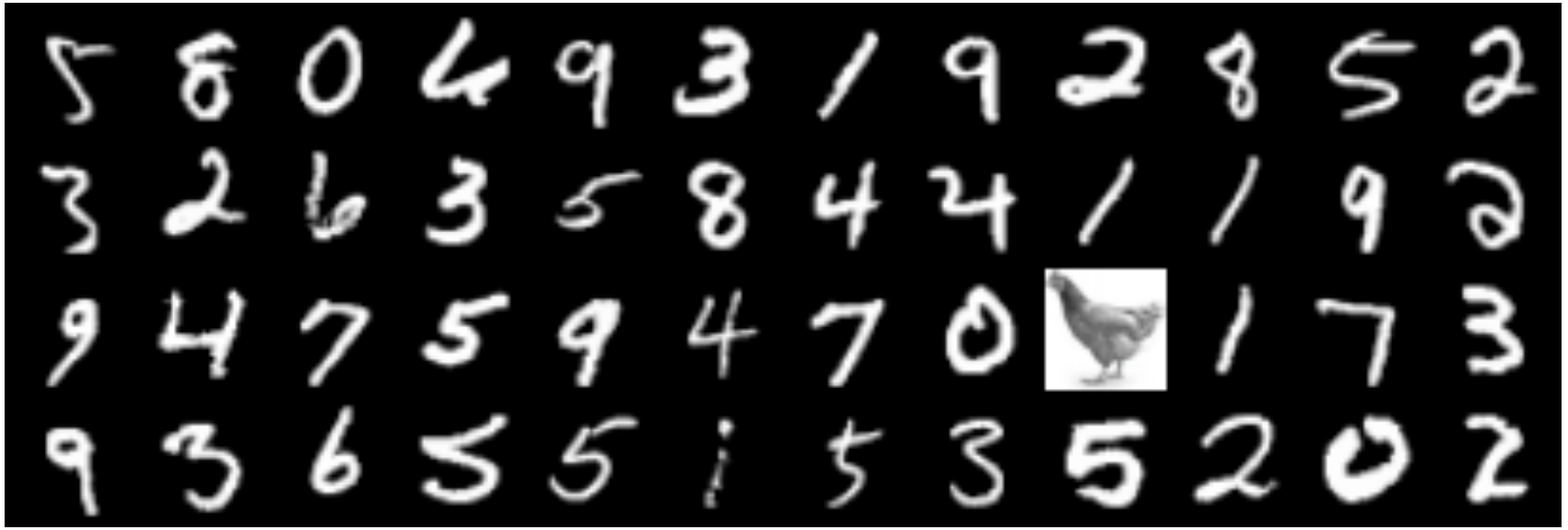


Image: emiliendupont blog

Source: https://emiliendupont.github.io/2018/03/14/mnist-chicken/

Confidence Mismatch

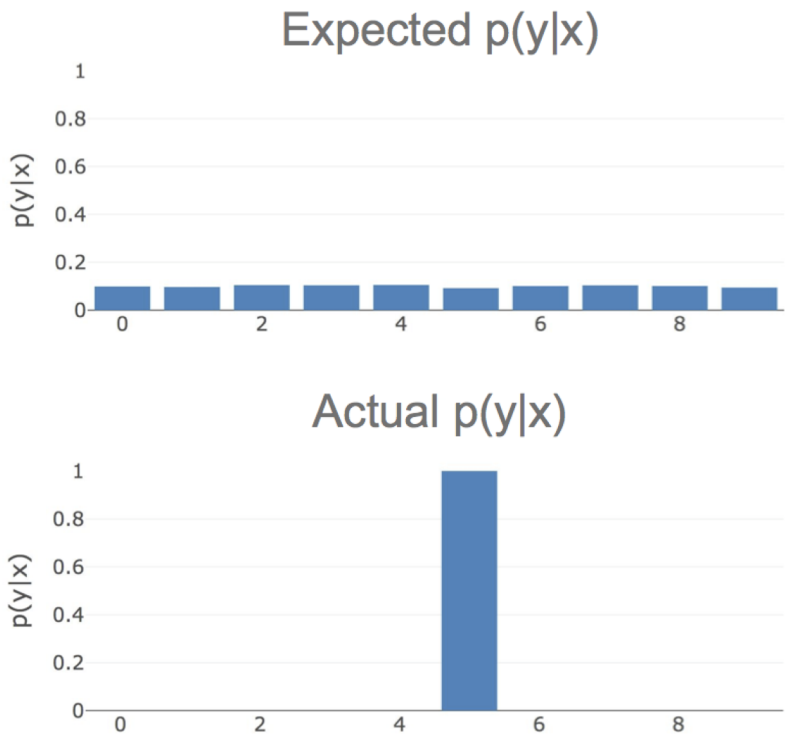


Image: emiliendupont blog

76.4% confident that it's a 5

99.9% confident that it's a 5

Image: emiliendupont blog

Source: https://emiliendupont.github.io/2018/03/14/mnist-chicken/

Use Autoencoder to identify training distribution

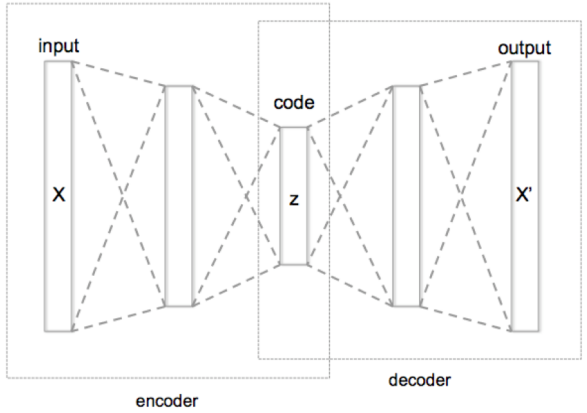


Image: Wikipedia, Chervinskii

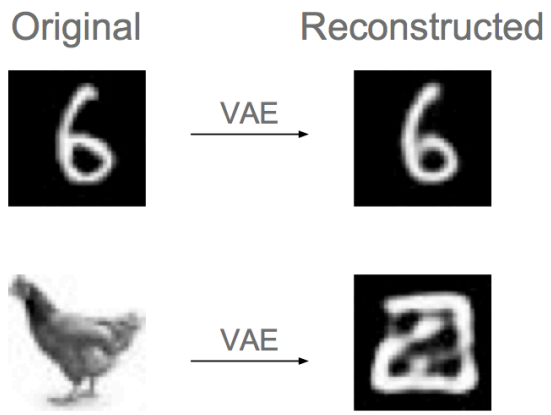


Image: emiliendupont blog

Source: Chervinskii / CC BY-SA 4.0

Source: https://emiliendupont.github.io/2018/03/14/mnist-chicken/

Finding Out-of-distribution Samples

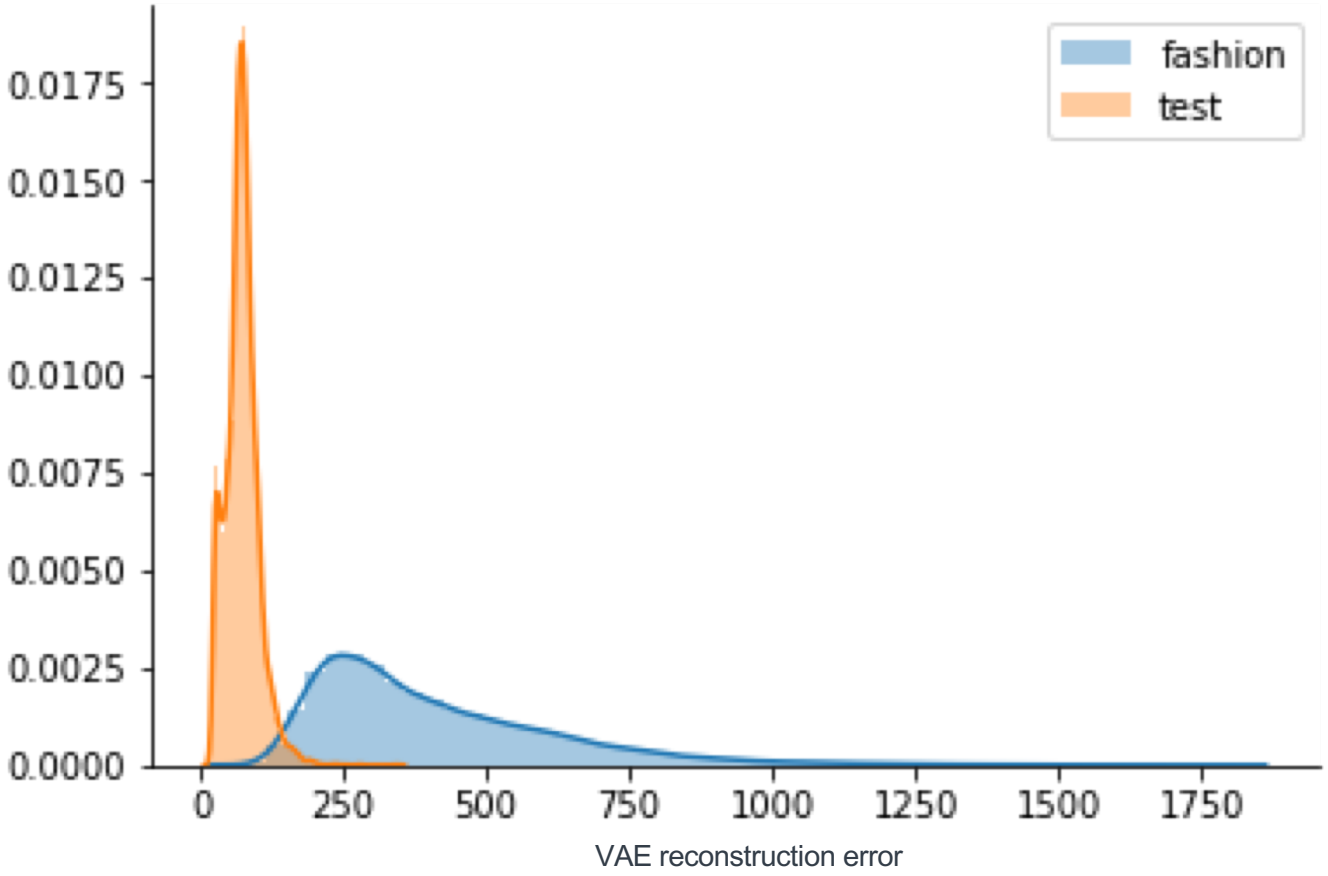
MNIST

Fashion MNIST

Image: emiliendupont blog

Source: https://emiliendupont.github.io/2018/03/14/mnist-chicken/

Separation of New Distribution (fashion) from Training Distribution



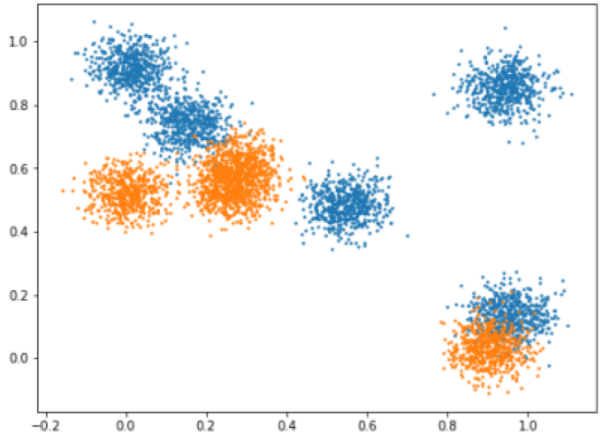
Recent Work on Uncertainty (Harang et al)

Harang, Richard, and Ethan M. Rudd. "Principled Uncertainty Estimation for Deep Neural Networks." *arXiv preprint arXiv:1810.12278* (2018).

Leverages Real-NVP

- Real Non-Volumetric Preserving Transformations
- Dinh, Laurent, Jascha Sohl-Dickstein, and Samy Bengio. "Density estimation using Real NVP." *arXiv preprint arXiv:1605.08803* (2016).

Captures both class-conditional densities as well as overall density (and hence overall uncertainty)



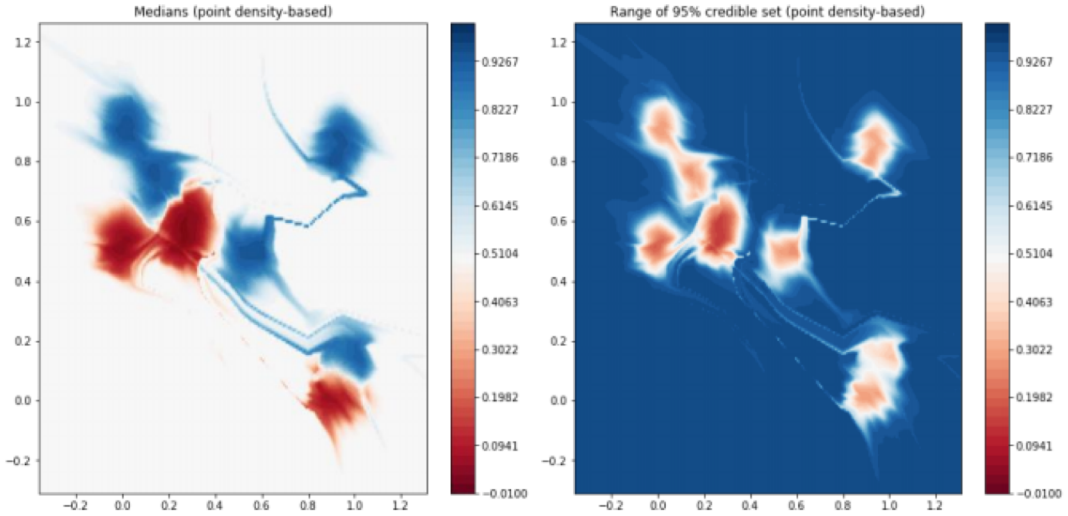


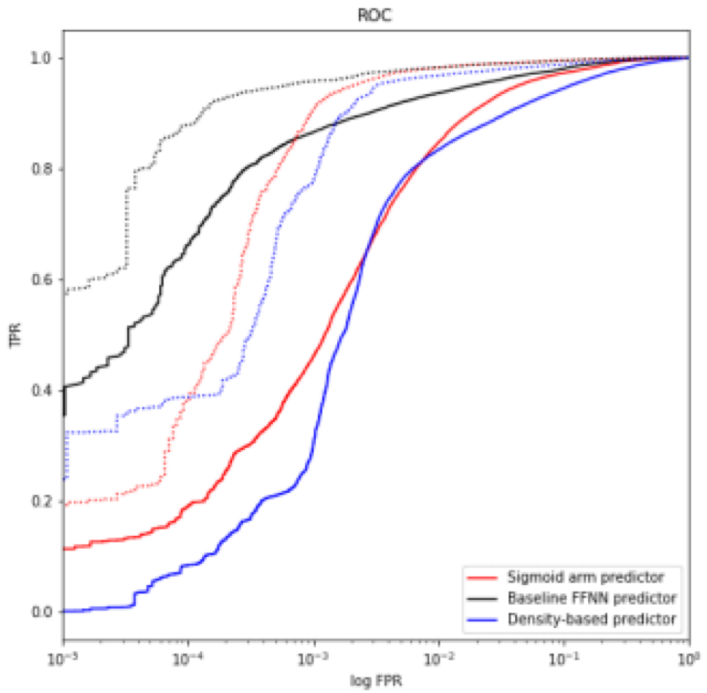
Image: Harang et al

Recent Work on Uncertainty, Continued (Harang et al)

Improvement in model performance based on removing samples with high uncertainty (dotted line)

- Especially effective at low false positive rates!

Real-NVP based selection



Raw NN based selection

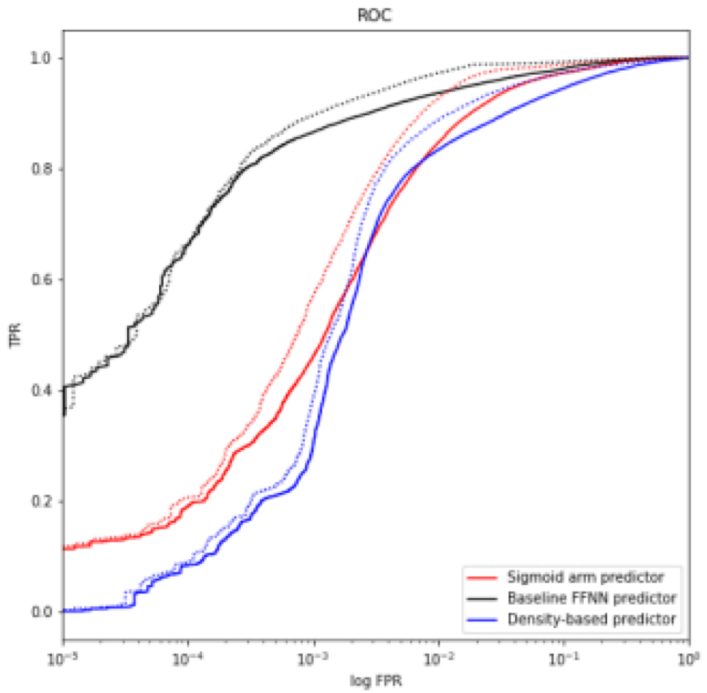
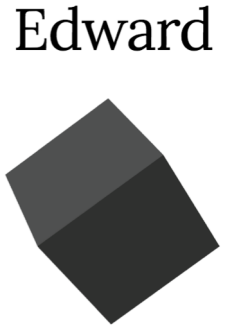


Image: Harang et al

Enablers:

PPLs (Probabalistic Programming Languages)



Automatic Differentiation Libraries

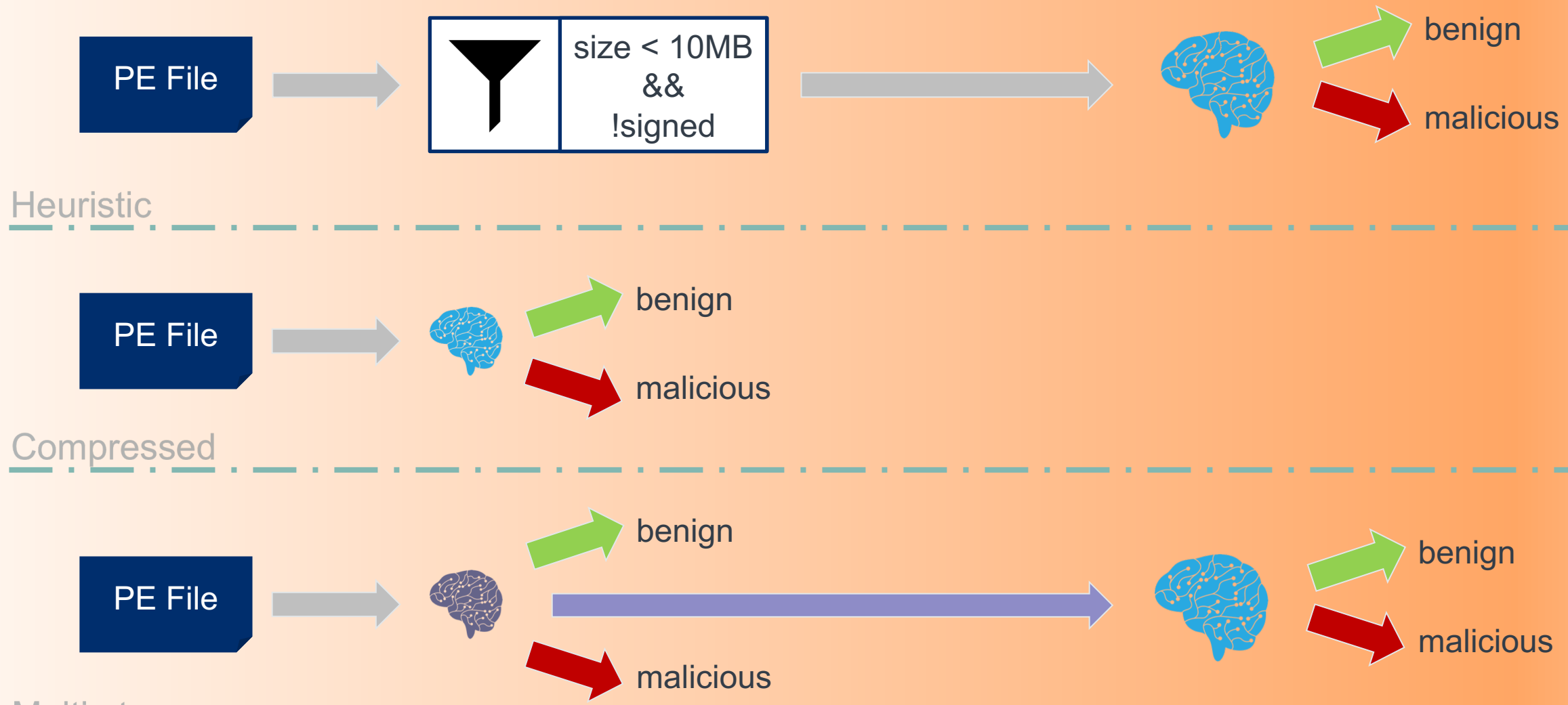
Confidence

Takeaway

Consider using autoencoding or similar technique to detect (and annotate) outside-of-training-distribution samples.

Keep an eye on probabilistic programming usability improvements. When suitable, adopt to improve models and provide valuable context about uncertainty.

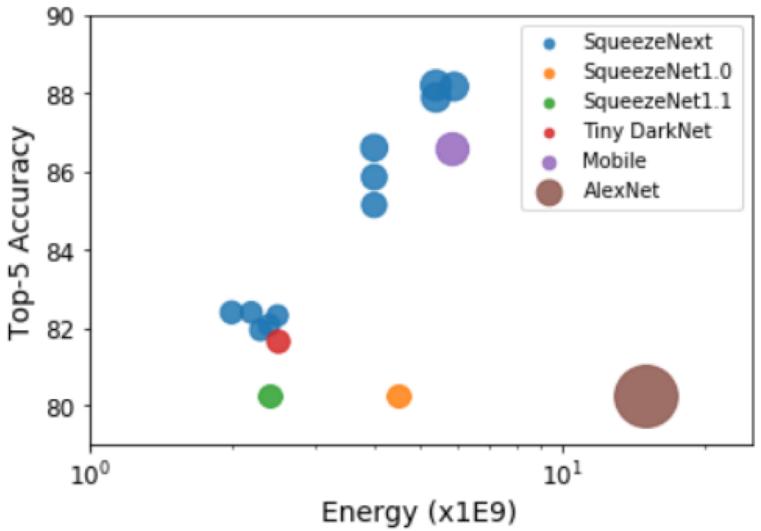
Architecture



Model Compression

Improves model size (memory), energy usage, speed, and sometimes accuracy

Can replace other models or be integrated in multistage



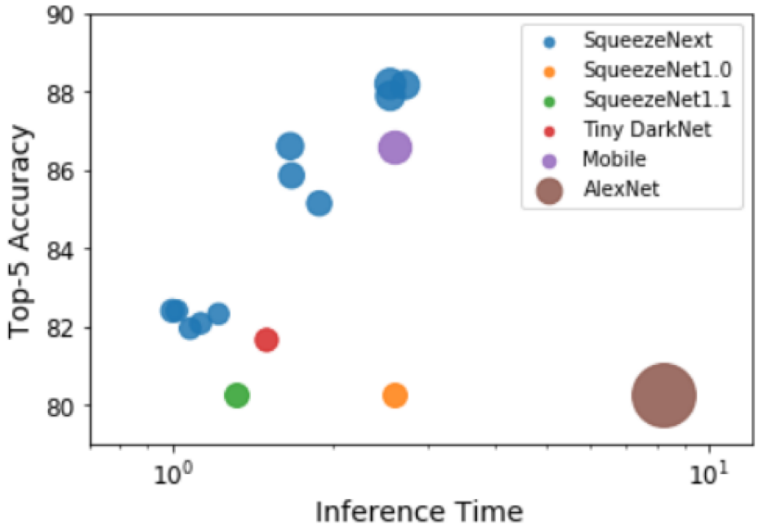
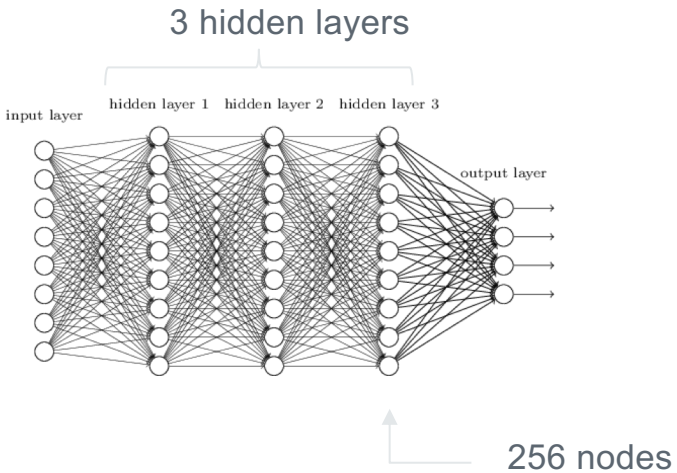


Image: Gholami et al

Ember



Objective:

Develop distributed malicious file classifier that minimizes costs while maintaining overall accuracy

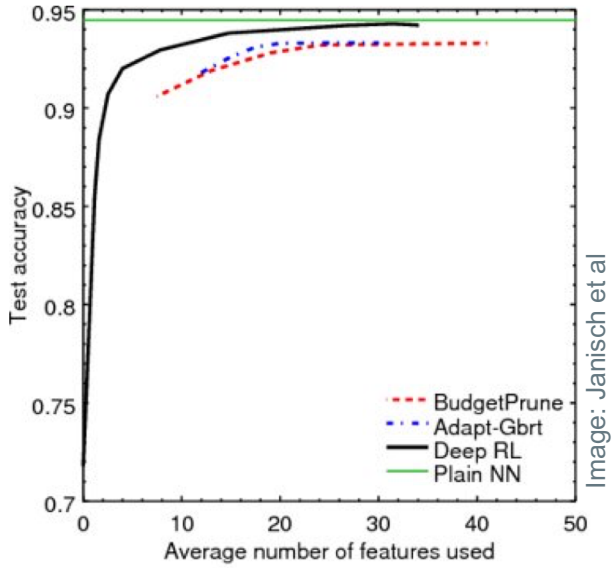
Constraints:

- Limit considered costs to: parsing, model runtime, transport
- Use Ember dataset
- Use standardized model architecture: DNN, fully connected 3 hidden layers, 256 nodes per hidden layer

Technique Selection

- Q-Learning Network
- Important approach: jointly modeling cost and performance
- Implemented Janisch et al 2017, “Classification with Costly Features using Deep Reinforcement Learning”
- Reinforcement learning approach with DNN as agent “brain”
  - Double Deep

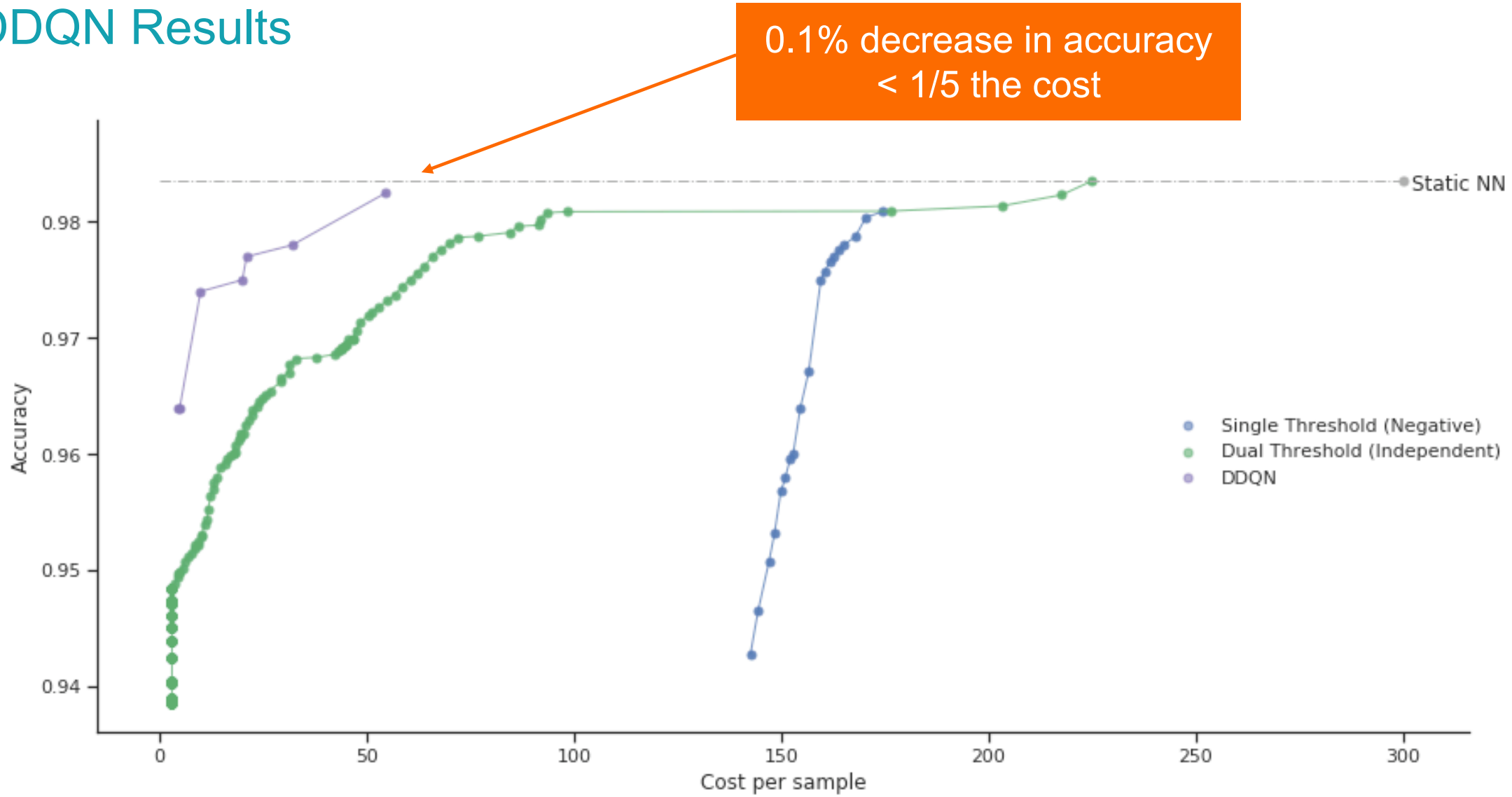
Image: Deepmind



(a) MiniBooNE

Image: Janisch et al

DDQN Results



Architecture

Takeaway

Architectural improvements can apply directly to scaling. If you are missing data due to heuristic filtering techniques used at the edge, consider expanding the scope of modeling efforts to the edge.

Review

Measurement

Explainability

Confidence

Architecture

Questions?

References

Alber, Maximilian, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T. Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. "iNNvestigate neural networks!." arXiv preprint arXiv:1808.04260 (2018).

Anderson, Hyrum S., and Phil Roth. "EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models." *arXiv preprint arXiv:1804.04637* (2018).

Dinh, Laurent, Jascha Sohl-Dickstein, and Samy Bengio. "Density estimation using Real NVP." arXiv preprint arXiv:1605.08803 (2016).

Gholami, Amir, Kiseok Kwon, Bichen Wu, Zizheng Tai, Xiangyu Yue, Peter Jin, Sicheng Zhao, and Kurt Keutzer. "SqueezeNext: Hardware-Aware Neural Network Design." arXiv preprint arXiv:1803.10615 (2018).

Harang, Richard, and Ethan M. Rudd. "Principled Uncertainty Estimation for Deep Neural Networks." arXiv preprint arXiv:1810.12278 (2018).

Janisch, Jaromír, Tomáš Pevný, and Viliam Lisý. "Classification with Costly Features using Deep Reinforcement Learning." arXiv preprint arXiv:1711.07364 (2017).

Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions." In Advances in Neural Information Processing Systems, pp. 4765-4774. 2017.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel et al. "Scikit-learn: Machine learning in Python." *Journal of machine learning research* 12, no. Oct (2011): 2825-2830.

Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization." In ICCV, pp. 618-626. 2017.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier." In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1135-1144. ACM, 2016.

Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." In European conference on computer vision, pp. 818-833. Springer, Cham, 2014.

References (Online)

https://www.endgame.com/blog/technical-blog/opening-machine-learning-black-box-model-interpretability

https://github.com/albermax/innvestigate

https://emiliendupont.github.io/2018/03/14/mnist-chicken/

Image References

Image: scikit-learn.org	https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html#sphx-glr-auto-examples-model-selection-plot-roc-py
Image: Endgame	https://github.com/endgameinc/ember/blob/master/resources/ember-notebook.ipynb
Image: Information Age	https://www.information-age.com/explainable-ai-123476397/
Image: DARPA	https://www.darpa.mil/program/explainable-artificial-intelligence
Image: Washington Times	https://www.washingtontimes.com/news/2018/jun/29/darpas-explainable-ai-a-common-sense-comfort-in-a-/
Image: TNW	https://thenextweb.com/artificial-intelligence/2018/02/27/bye-bye-black-box-researchers-teach-ai-to-explain-itself/
Image: LIME	https://homes.cs.washington.edu/~marcotcr/blog/lime/
Image: SHAP	https://slundberg.github.io/shap/notebooks/Census%20income%20classification%20with%20LightGBM.html
Image: albermax	https://github.com/albermax/innvestigate
Image: Selvaraju et al	Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization." In ICCV, pp. 618-626. 2017.
Image: Endgame blog	https://www.endgame.com/blog/technical-blog/opening-machine-learning-black-box-model-interpretability
Image: emiliendupont blog	https://emiliendupont.github.io/2018/03/14/mnist-chicken/
Image: Wikipedia, Chervinskii	https://en.wikipedia.org/wiki/Autoencoder
Image: Harang et al	https://arxiv.org/abs/1810.12278 https://www.camlis.org/richard-harang
Image: Google	https://blog.google/products/google-vr/google-lens-real-time-answers-questions-about-world-around-you/
Image: Gholami et al	Gholami, Amir, Kiseok Kwon, Bichen Wu, Zizheng Tai, Xiangyu Yue, Peter Jin, Sicheng Zhao, and Kurt Keutzer. "SqueezeNext: Hardware-Aware Neural Network Design." arXiv preprint arXiv:1803.10615 (2018).
Image: Deepmind	https://deepmind.com/blog/
Image: Janish et al	Janisch, Jaromír, Tomáš Pevný, and Viliam Lisý. "Classification with Costly Features using Deep Reinforcement Learning." arXiv preprint arXiv:1711.07364 (2017).