BelPy

Towards a more transparent AI - Decrypting ML models using LIME

Laisha Wadhwa

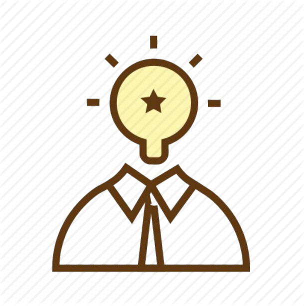
Session Agenda

Introduction

Importance of interpretable AI

LIME: What and How?

Model Interpretation examples



Final thoughts

About Me

@laishawadhwa

/laishawadhwa

- ➔ Data Engineer, Couture.ai
- ➔ Microsoft AI challenge 2018 winner
- ➔ Amex & Techgig Geek Goddess 2019 AIML hackathon winner
- ➔ Sabre Hack 2019 Winner
- ➔ Mercedes Benz Digital Challenge winner
- ➔ Icertis AI and blockchain RU
- ➔ Women Tech Network, Global Ambassador
- ➔ Podcast Host: Co-Learning Lounge
- ➔ Technical Writer, Omdena
- ➔ Tech Speaker

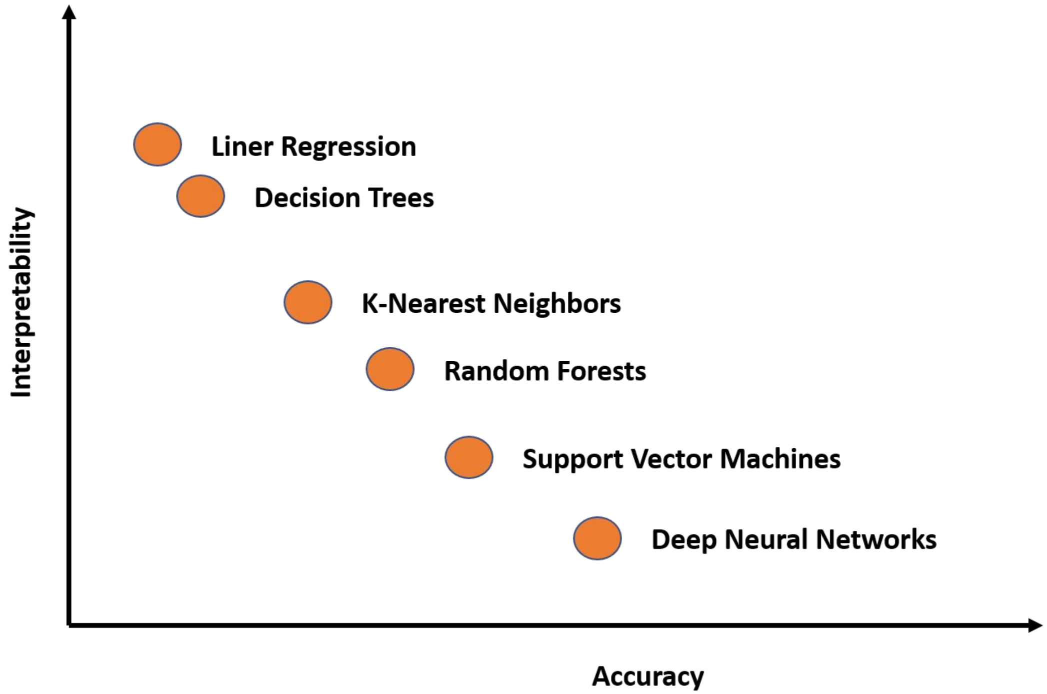
We all have been here!

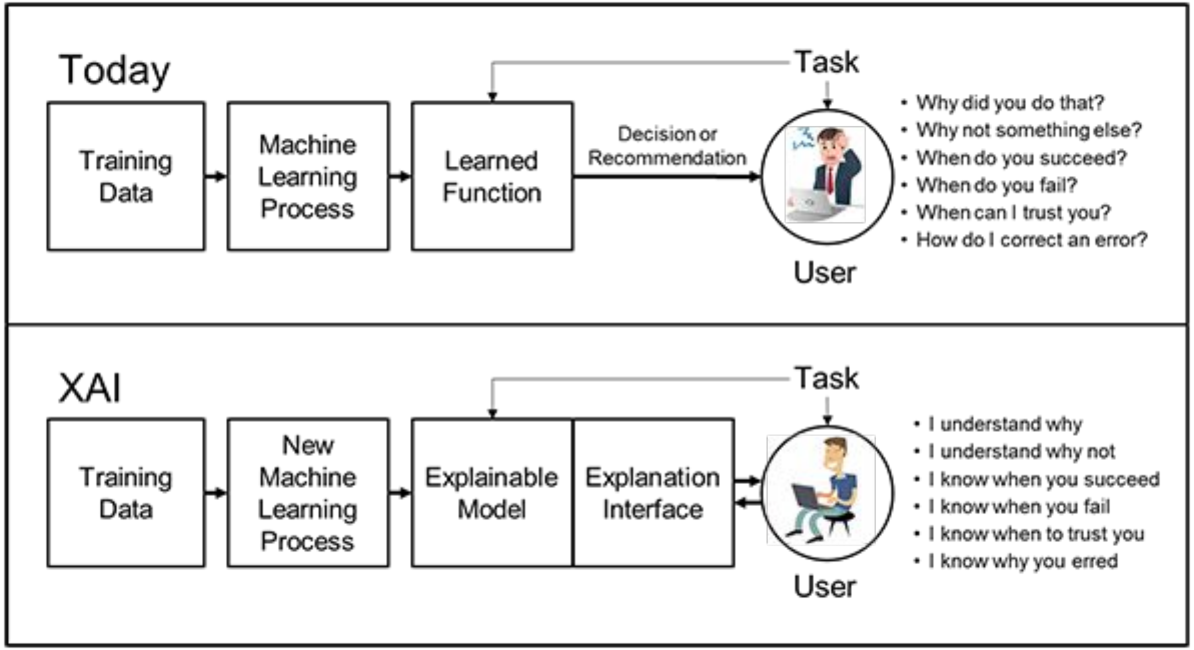
When your classifier achieves 98% accuracy

Your classifier's Output

YOU

Accuracy vs Interpretability

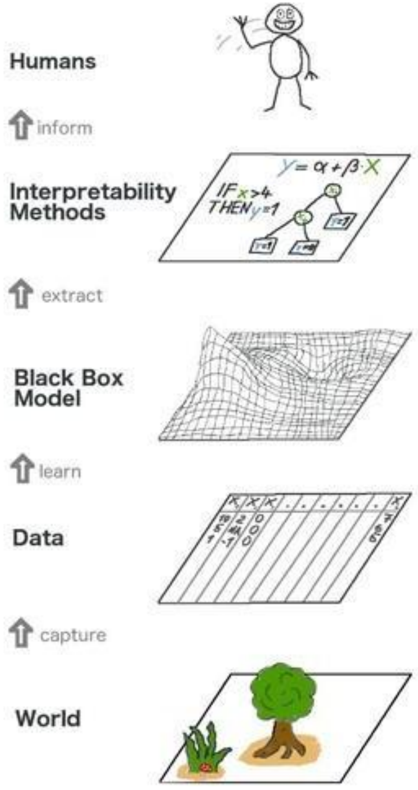




XAI: **E**xplainable **A**rtificial **I**ntelligence

Source: https://www.darpa.mil/program/explainable-artificial-intelligence

ML and DL models have become a black-box



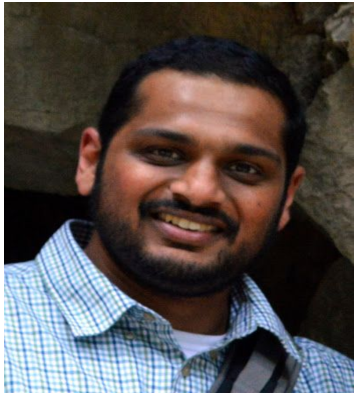
Here, LIME comes into picture!

The idea is to interpret the models and understand why the classifier chose a particular class or why the model gives more False Positives ??

Why is it important?

Creators of LIME

Riberio Marco



Sameer Singh

Guestrin Carlos

What is LIME??

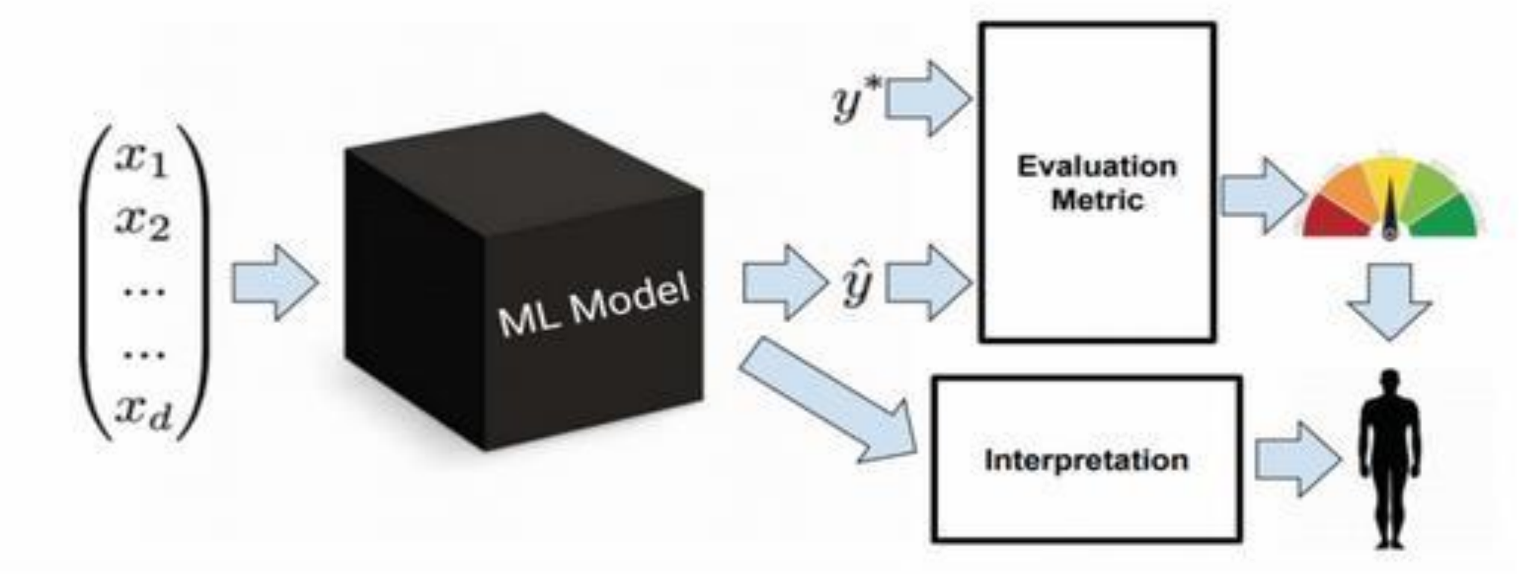
Local

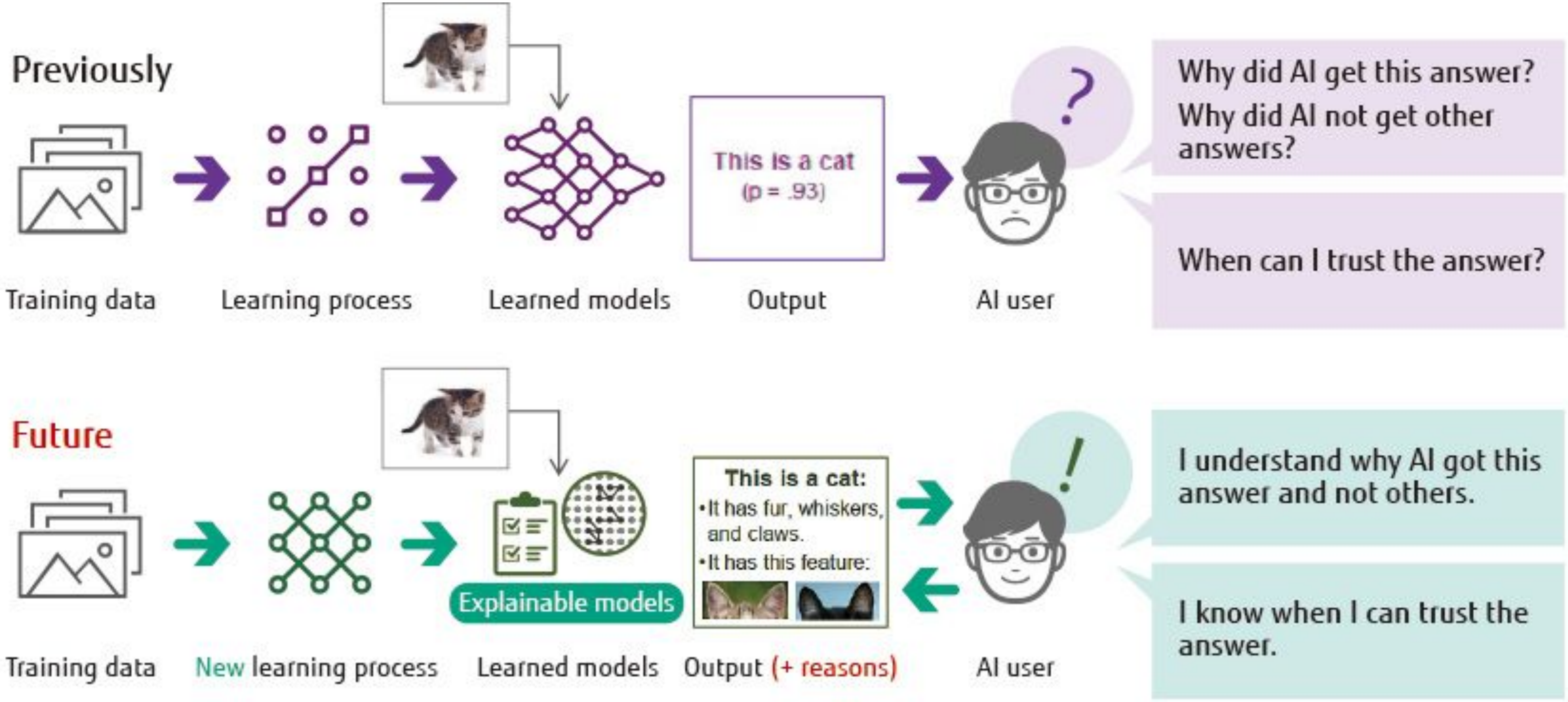
Interpretable

Model agnostic

Explanations

LIME helps to interpret models, HOW?




Previously
Training data
Learning process
Learned models
This is a cat
(p = .93)
Output
AI user
Why did AI get this answer?
Why did AI not get other answers?
When can I trust the answer?
Future
Training data
New learning process
Explainable models
Learned models
This is a cat:
•It has fur, whiskers, and claws.
•It has this feature:
Output (+ reasons)
AI user
I understand why AI got this answer and not others.
I know when I can trust the answer.

Interpretable Machine Learning with LIME

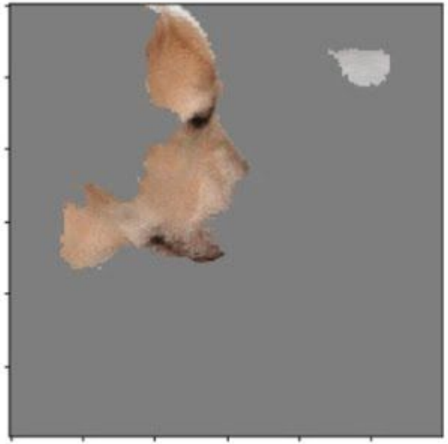
Machine Learning Model

This is a "labrador"

Why? →

LIME

Because:



Learn another model on top of the ML Model

Generate new data by permutation of the input data

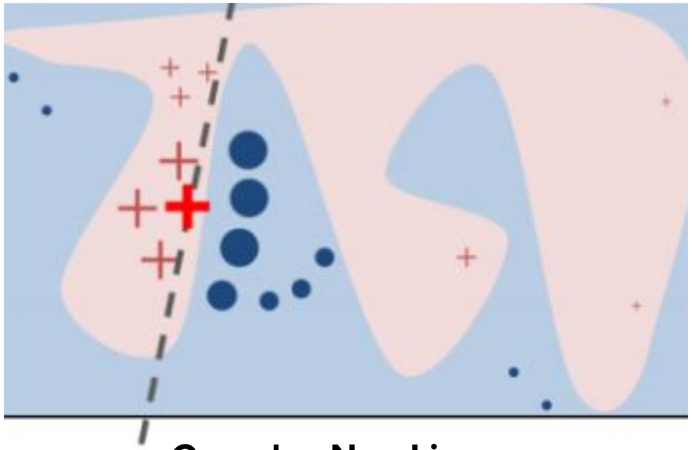
Prediction on this new data on the the simple (local fit: an approximation)

More weight is being put on data that is similar to the original data)

Using LIME to explain a given model M’s prediction using $\mathbf{x}_i$

1. Sample points around $\mathbf{x}_i$.
2. Get the predictions for observation $\mathbf{x}_i$ using the given model M.
3. Weigh samples according to distance from $\mathbf{x}_i$.
4. Learn new simple model on the weighted samples.
5. Use the simple model to explain.

Global



Complex Non Linear

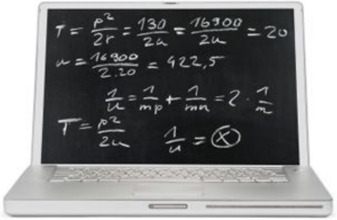
Local Simple Non Linear

The Secret Sauce

- LIME minimises the **locality-aware loss** which is measured using a mathematical formula (It's basic math!) which takes into consideration measure of **unfaithfulness of the model** in approximating the target variable and measure of model complexity (ρ) of explanation.

- E.g. if the model to be explained is decision tree, ρ can be depth of the tree or in case of linear explanation models it can be number of non zero weights.

The Math - locality-aware loss



$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \; \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

First Term : measure of unfaithfulness of g in approximating f in the locality defined by Pi. This is termed as **locality-aware loss** in the original paper

Last term : measure of model complexity of explanation g. For example if your explanation model is decision tree it can be depth of the tree or in case of linear explanation models it can be number of non zero weights

Why should you use it?

<u>LIME</u>

> Interpretability.

>Explains decision boundaries of the model in human understandable form.

> Local Fidelity &Model Agnostic

> Locally approximate global model

<u>Traditional Approach</u>

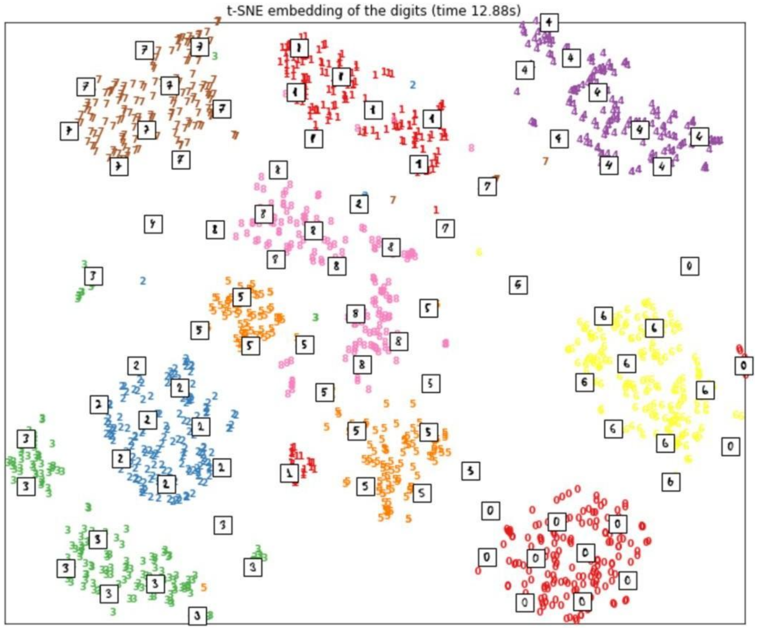
> EDA: Visualisation [t-SNE

> Model Performance Evaluation Metrics.

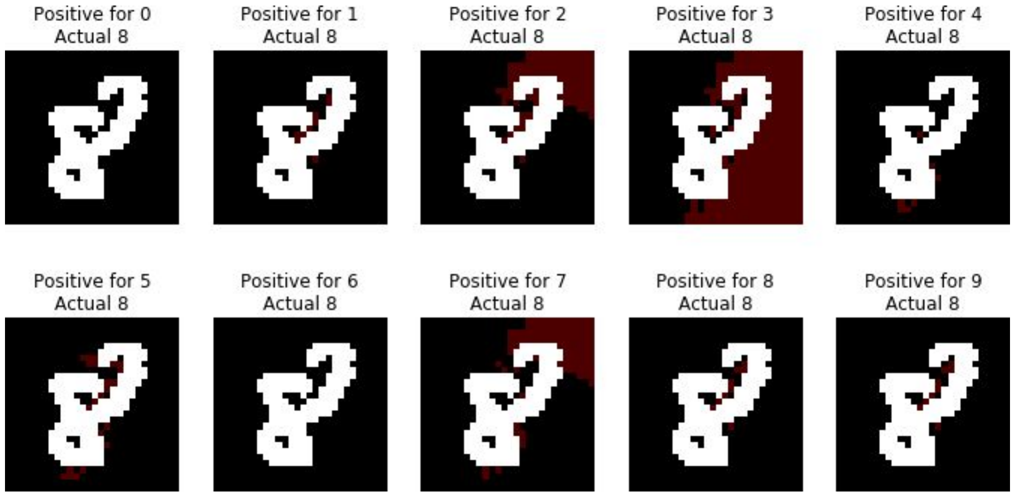
> Can’t comprehend the decision boundary with changing nature of input points.

> No insight on feature imp for new data pts.

Traditional



LIME



What all can LIME do?

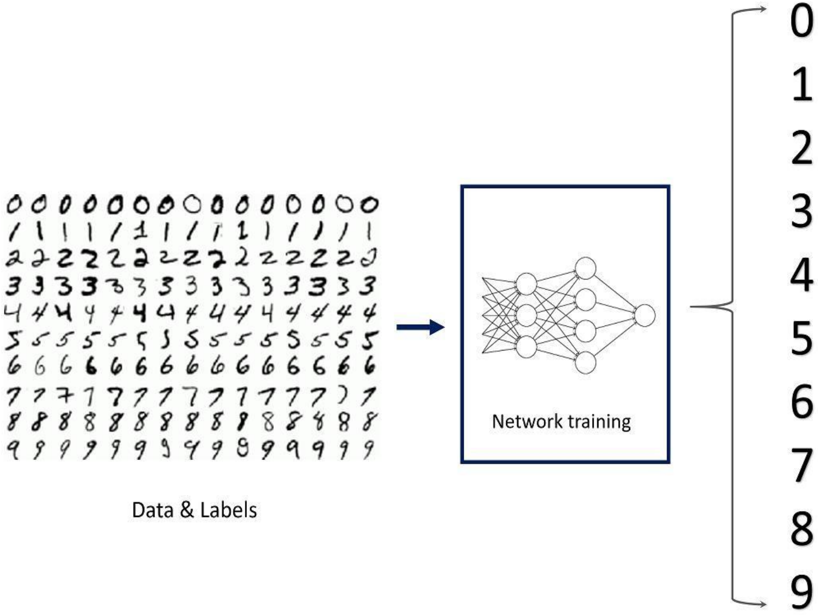
- It can explain models with any kind of data:

a) **Text** : It represents presence/absence of words. **LimeTextExplainer**

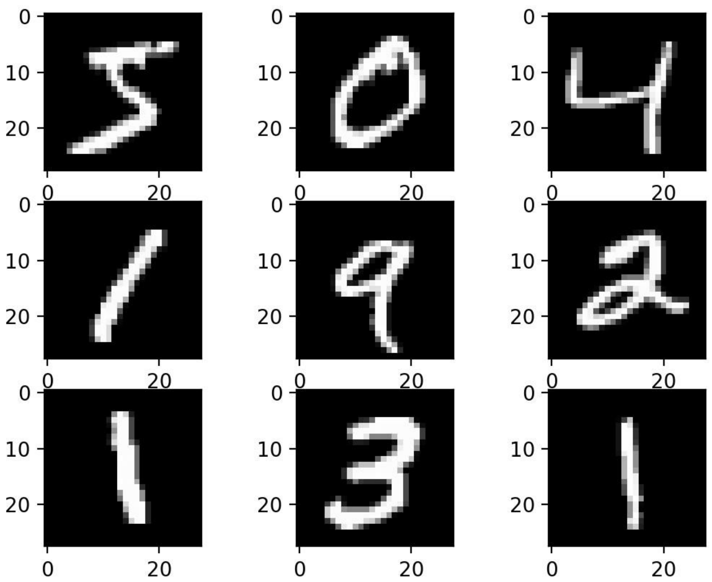
b) **Image** : It represents presence/absence of super pixels (contiguous patch of similar pixels). **LimeImageExplainer**

c) **Tabular data** : It is a weighted combination of columns. **LimeTabularExplainer**

DIGIT classification



Data & Labels



Link for Colab Notebooks and Slide deck

https://github.com/laishawadhwa/BelPy-2021

How do we do it??

```
simple_dt_pipeline = Pipeline([
    ('Make Gray', makegray_step),
    ('Flatten Image', flatten_step),
    ('DT', DecisionTreeClassifier(criterion = "gini",
            random_state = 100,max_depth=7, min_samples_leaf=5) )
                         ])
```

LIME: explain_instance and ImageExplainer

```
explainer = lime_image.LimeImageExplainer(verbose = False)
segmenter = SegmentationAlgorithm('quickshift', kernel_size=1, max_dist=200, ratio=0.2)
testIndex = 7
explanation = explainer.explain_instance(X_test[testidx],
                                         classifier_fn = simple_dt_pipeline.predict_proba,
                                         top_labels=10, hide_color=0, num_samples=10000,
segmentation_fn=segmenter)
```

- LIME tests out what happens to your black box model's predictions when you feed variations or perturbations of your dataset.
- How? By generating a new set comprising of perturbed samples and the corresponding black box model's predictions.
- LIME then trains an interpretable model weighted by the proximity of the sampled instances to the instance of interest.

Let's see how well can a decision tree differentiate the numbers!!

Let’s delve deeper

Let’s say we are trying to classify wolves and dogs.

Wolf

Dog

What if we have wolves and dogs in entirely different backgrounds? This is entirely possible as wolves are mostly found in the wild (in snow, jungles, etc.) while dogs are in completely different backgrounds (households) generally.

We observe that, our model is actually learning the background pretty well!!

Let’s see this in action! Does it work well for DL?]

Image classification- How to?

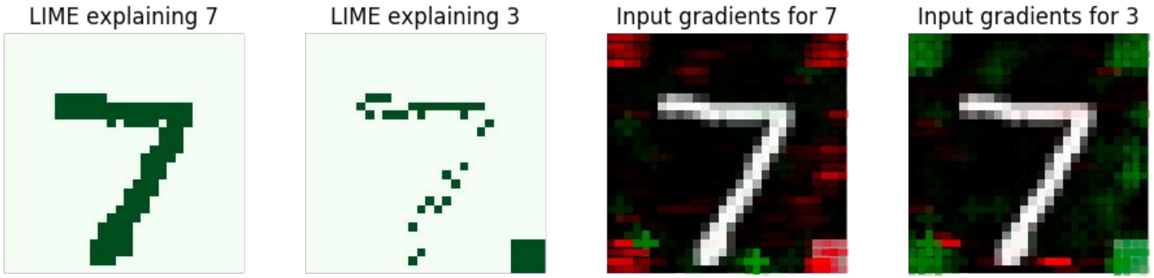
```
#Loading model
checkpoints_dir = '/content/tf-models/slim/pretrained'
init_fn = slim.assign_from_checkpoint_fn(
    os.path.join(checkpoints_dir, 'inception_v3.ckpt'),
    slim.get_model_variables('InceptionV3'))
init_fn(session)
```

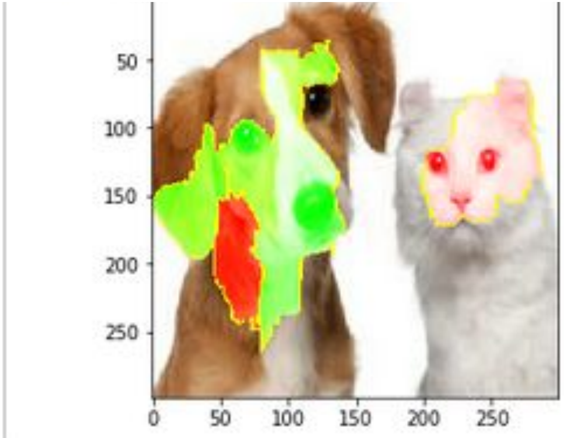
Running inference on a set of images

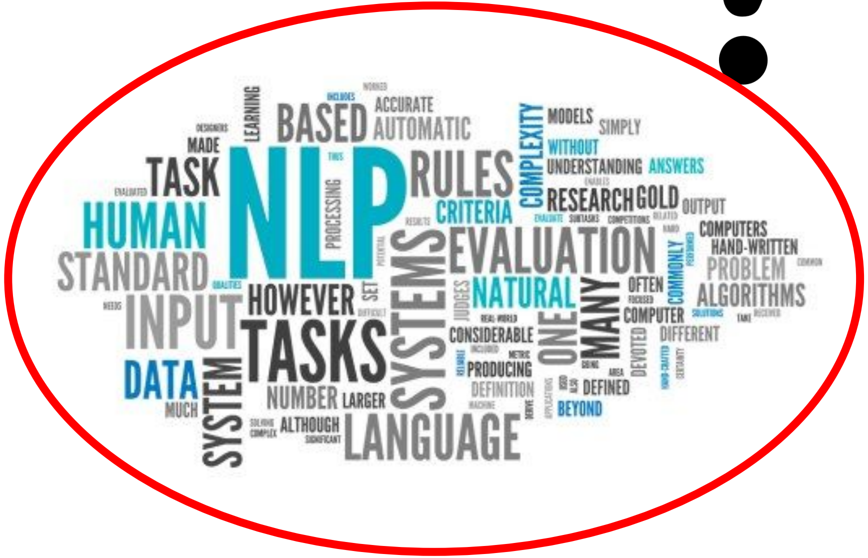
```
#prediction Function for running inference

def predict_fn(images):
    return session.run(probabilities, feed_dict={processed_images: images})

preds = predict_fn(images)
for x in preds.argsort()[0][-5:]:
    print (x, names[x], preds[0,x])
```

Which LIME class to use?

```
explainer = lime_image.LimeImageExplainer()
explanation = explainer.explain_instance(image, predict_fn,
                                         top_labels=10, hide_color=0, num_samples=1000)
```


LIME explaining 7
LIME explaining 3
Input gradients for 7
Input gradients for 3


50
100
150
200
250
0
50
100
150
200
250


NLP
RULES
EVALUATION
SYSTEMS
TASKS
LANGUAGE
HUMAN
STANDARD
INPUT
DATA
SYSTEM
TASK
BASED
AUTOMATIC
ACCURATE
CRITERIA
NATURAL
HOWEVER
MANY
ONE
RESEARCH
GOLD
COMPLEXITY
MODELS
SIMPLY
WITHOUT
UNDERSTANDING
ANSWERS
OUTPUT
COMPUTERS
HAND-WRITTEN
PROBLEM
ALGORITHMS
COMMONLY
OFTEN
COMPUTER
DIFFERENT
CONSIDERABLE
PRODUCING
DEFINITION
DEFINED
BEYOND
NUMBER
LARGER
ALTHOUGH
MUCH
MADE
LEARNING
PROCESSING
DEVOTED

We have seen digits, cats and dogs..

Let’s see Text classification on stackoverflow post and tags dataset

	post	tags
0	what is causing this behavior in our c# datet...	c#
1	have dynamic html load as if it was in an ifra...	asp.net
2	how to convert a float value in to min:sec i ...	objective-c
3	.net framework 4 redistributable just wonderi...	.net
4	trying to calculate and print the mean and its...	python

How will we do it?

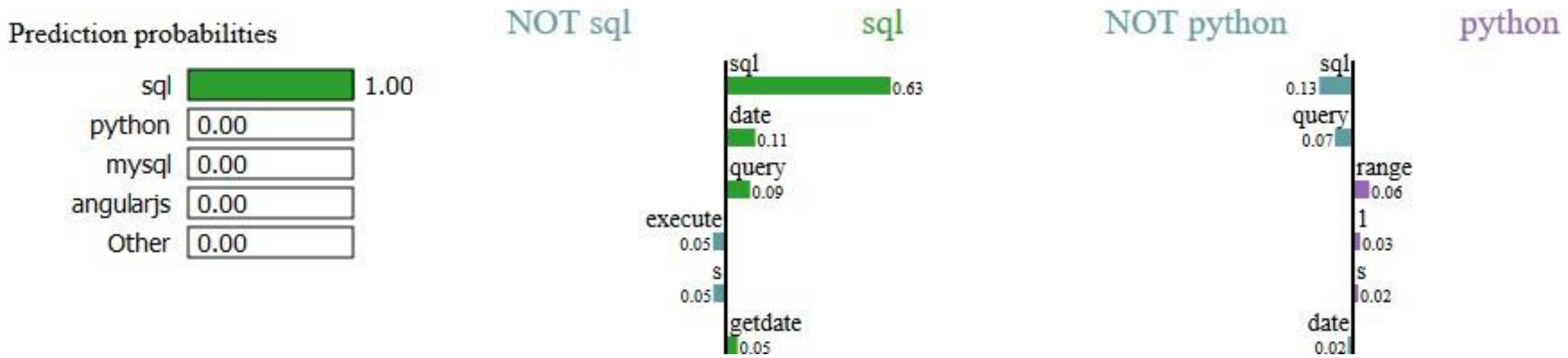
- We’ll first convert words to vector using CountVectorizer()

Next , we build a Logistic regression model for text data

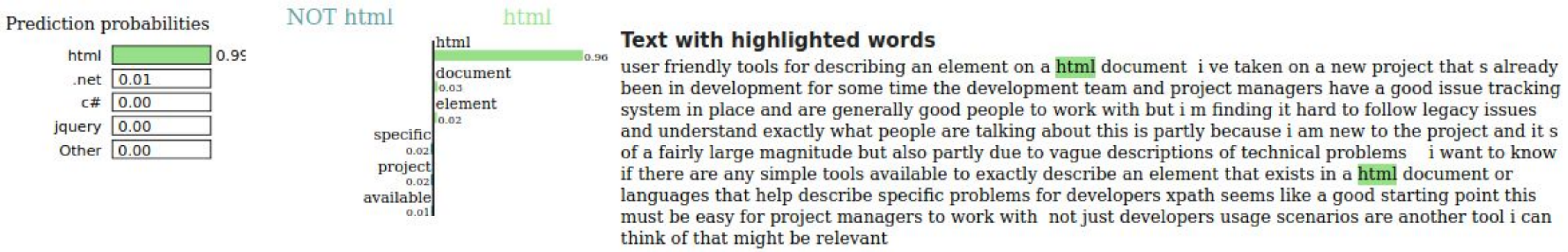
```
Vectorizer = CountVectorizer(analyser='word', token_pattern=r'\w(1,)', ngram_range=(1, 3), stop_words =
'english', binary=True)  train_vectors = vectorizer.fit_transform(X_train)
#iniatializing the model
logreg = LogisticRegression(n_jobs=1, C=1e5)

#fitting the model with train_vectors
logreg.fit(train_vectors, y_train)
```

LimeTextExplainer

```
'''
  LimeTextExplainer
  LimeTextExplainer(kernel_width=25, verbose=False, class_names=None, feature_selection=u'auto',
split_expression=u'\W+', bow=True)
  Explains text classifiers. It uses an exponential kernel on cosine distance,
'''
explainer = LimeTextExplainer(class_names=class_names)

'''
   explain_instance(text_instance, classifier_fn, labels=(1, ), top_labels=None, num_features=10,
num_samples=5000, distance_metric=u'cosine', model_regressor=None)
'''

exp = explainer.explain_instance(X_test[idx], c.predict_proba, num_features=6, labels=[5, 3])
```


Prediction probabilities
sql 1.00
python 0.00
mysql 0.00
angularjs 0.00
Other 0.00
NOT sql
sql
sql 0.63
date 0.11
query 0.09
execute 0.05
s 0.05
getdate 0.05
NOT python
python
sql 0.13
query 0.07
range 0.06
1 0.03
s 0.02
date 0.02

Highlight the words for more context



Things to Explore

- You can check out similar library for interpretability called: SHAP (https://shap.readthedocs.io/en/latest/)

- **A Guide for Making Black Box Models Explainable** by **Christoph Molnar** is a great book to refer to if you want to delve deep and make machine learning more interpretable.

Conclusion

- Trust
- Transparency
- Public perception
- Improvement through feedback
- Making informed decisions
- Explanations are short, selective and possibly contrastive.

- Sampling from a Gaussian distribution.
- Complexity of the explanation model has to be defined in advance.

What have we learnt?

- ➔ Don’t trust models without explanations !
- ➔ Interpreting the machine’s intelligence is tough (For Humans as well).
- ➔ For Business centric use cases we must ensure that model is learning the scene description well irrespective of the accuracy. (Dog wolf example!).

The Future of Interpretability

- ✔ The focus will be on model-agnostic interpretability tools.
- ✔ Machine learning will be automated and, with it, interpretability.
- ✔ We do not analyze data, we analyze models.

Test whether A or B is better: For this we can also use partial dependence functions. What we do not have yet are statistical tests for arbitrary black box models.

- ✔ The data scientists will automate themselves.
- ✔ Interpretability could boost machine intelligence research.

References

1. “Why Should I Trust You?” Explaining the Predictions of Any Classifier — Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin.
2. Blog by the authors, https://homes.cs.washington.edu/~marcotcr/blog/lime/
3. Github link : https://github.com/marcotcr/lime:
4. Link to original paper : https://arxiv.org/abs/1602.04938

Questions?

@laishawadhwa
/laishawadhwa

Thank You!