Decrypting a Machine Learning model

By: Oriana Oniciuc
February 2020

[IASI AI]

Agenda

- Theoretical Concepts
- Demo
- Discussions

[IASI AI]

About us

[IASI AI] e·on

E.ON Software Development (ESD) is an E.ON group initiative. We are an Agile Delivery Facility that mainly focuses in insourcing strategic international and local developments and securing E.ON Group competitive business knowledge.

Engineering Services

- Software Development
- Software Testing
- Solutions Architecture
- Business Analysis
- Operations Support
- Product Management

Big Data and Analytics

Companies often need to analyze huge amounts of data in a short period of time. Big data is the centerpiece for digital business transformation — making customer interaction more impactful and answering critical business questions, highlighting areas for cost efficiency and opportunities for growth..

AI & ML

AI and ML have equally been given the same importance as the discovery of electricity at the beginning of Industrial Revolution. These frontier technologies, just like electricity, have ushered in a new era in the history of Information Technology.

Internet of Things

IoT is changing the way the world interacts with itself. Capitalize on IoT by investing in application solutions and connected products that enable disruptive, relevant, and frictionless user engagements.

PMO

Defines and maintains standards for project management within the organization. The Project Management Office provides guidance and standards in the execution of projects..

[IASI AI] e·on

Text Analytics

Analysing feedback from clients to improve the company's strategies.

Smart RPA

Creating automated solutions that can handle knowledge based decision making and unstructured data.

Knowledge management

Providing a feed of latest news in the industry, based on custom profiles.

Predictive Maintenance

Analytical service that predicts future outages in different components of the electricity Grid.

Disaggregation

Getting more insights of electricity usage, based on Smart Meters IoT data.

[IASI AI] e·on

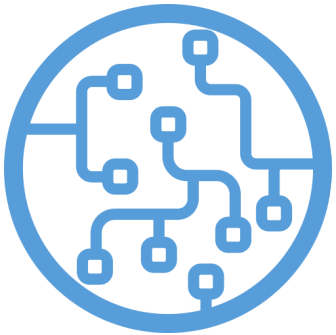

Azure
aws
docker
kubernetes
GitLab
splunk>
sonarqube
Jenkins
Scala
TypeScript
node
JS
python
Java
powercloud
APACHE
kafka
SAP
Atlassian
+ableau
scikits
learn
salesforce
Power BI
Gentrack

[IASI AI] e·on

Why Should You Care About Machine Learning Interpretability?

How to build an application with Machine Learning?

Data



Machine Learning model

Predictions & Deployment

How are we sure it is working?

Machine Learning model

Learn model

Trust model

Deploy model

Predictions & Deployment

Is it accuracy enough?


POLICE
"Does your car have any idea why my car pulled it over?"
PAUL NOTH

Definition **Interpret** means to explain or to present in understandable terms.

In the context of *Machine Learning systems*, we define **interpretability** as the ability to explain or to present in understandable terms to a human.

Social Motivation: Interpretability plays a critical role in the increased convenience, automation, and organization in our day-to-day lives promised by AI.

Commercial Motivation: Interpretability is required for regulated industry to adopt machine learning.

- Check and balance against accidental or intentional discrimination.
  - "Right to explanation."
- Hacking and adversarial attacks.
- Improved revenue

What is interpretability?

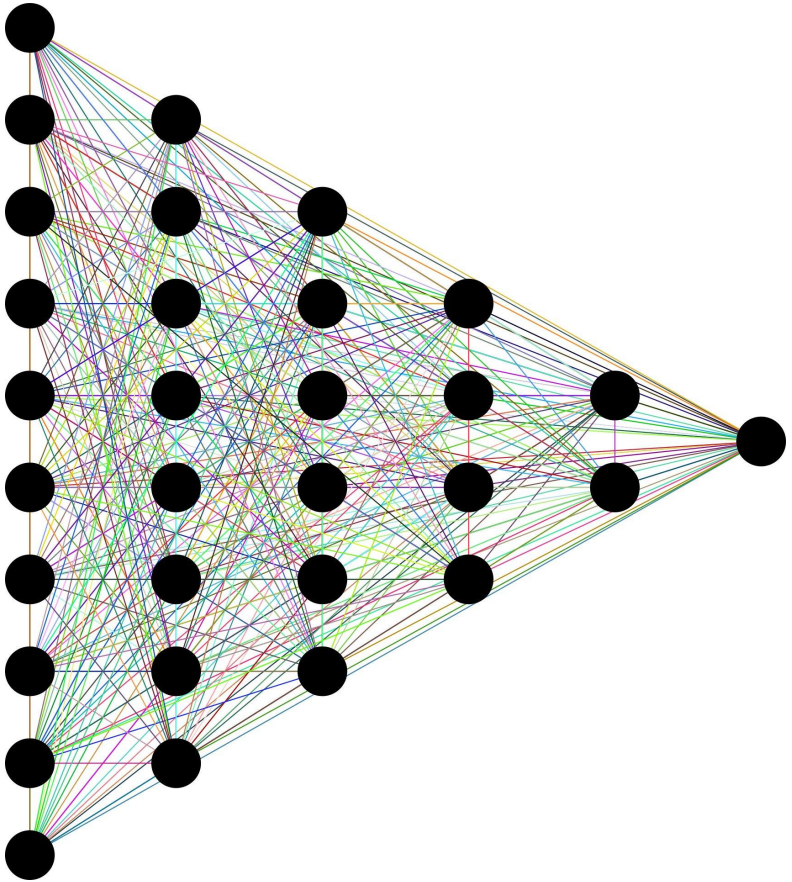
- Trust the AI system
- Make better decisions
- Improve the model

[IASI AI]

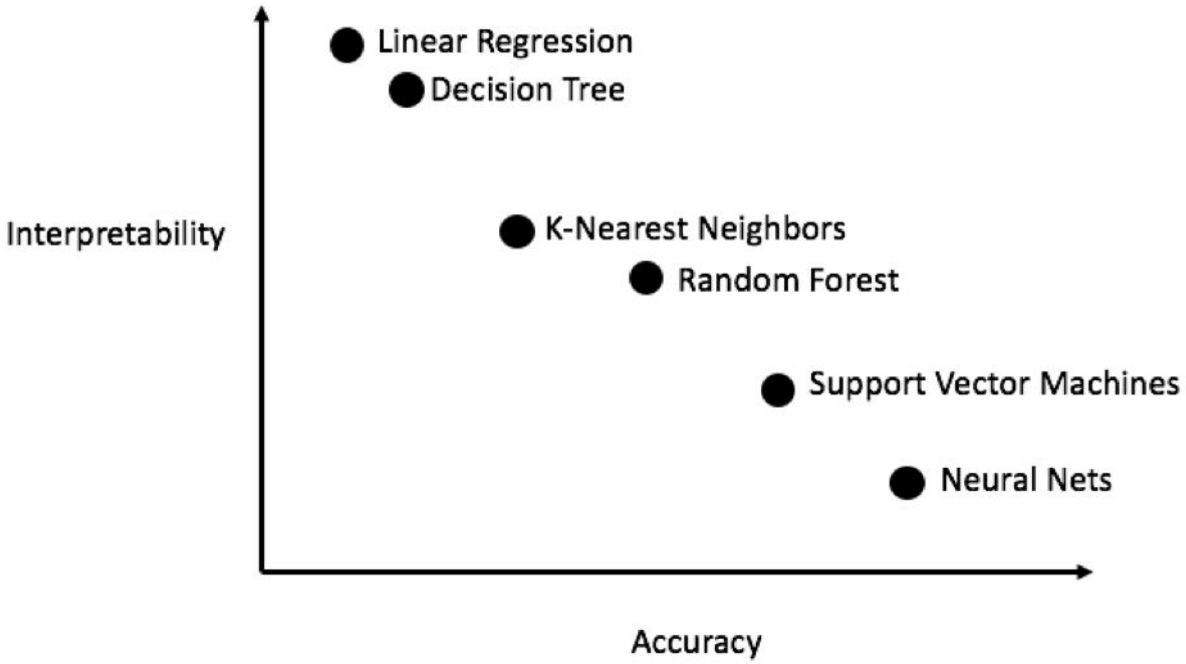
Types of models

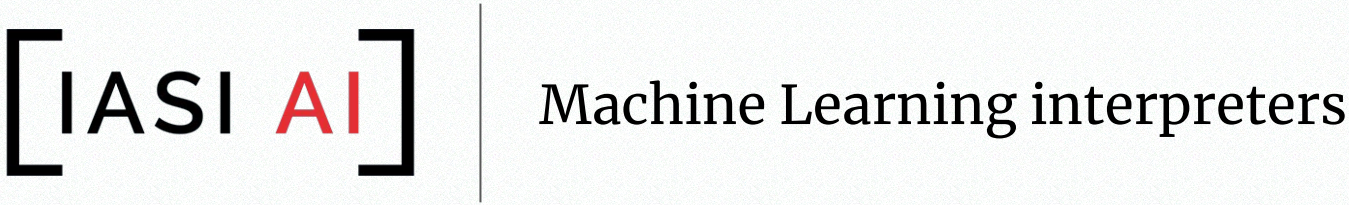
Algorithm	Linear	Monotone	Interaction	Task
Linear regression	Yes	Yes	No	regr
Logistic regression	No	Yes	No	class
Decision trees	No	Some	Yes	class,regr
RuleFit	Yes	No	Yes	class,regr
Naive Bayes	No	Yes	No	class
k-nearest neighbors	No	No	No	class,regr

Uninterpretable models - “Black Box”



The choice is yours!




IASI AI
Machine Learning interpreters

- Surrogate model
  - Simple model out of fancy model

- Tree interpreter
  - Gives feature importance
  - Creates global picture of the effect of features

Desirable aspects of a **model-agnostic explanation system** are:

- **Model flexibility**: The interpretation method can work with any machine learning model, such as random forests and deep neural networks.

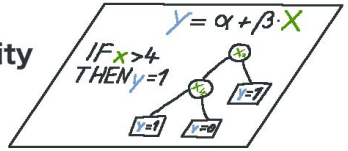
- **Explanation flexibility**: You are not limited to a certain form of explanation. In some cases it might be useful to have a linear formula, in other cases a graphic with feature importances.

- **Representation flexibility**: The explanation system should be able to use a different feature representation as the model being explained. For a text classifier that uses abstract word embedding vectors, it might be preferable to use the presence of individual words for the explanation.

Humans

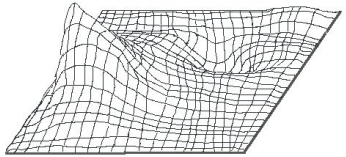
inform

Interpretability Methods



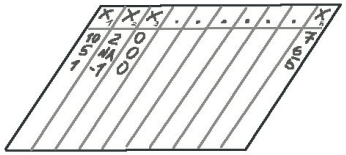
extract

Black Box Model



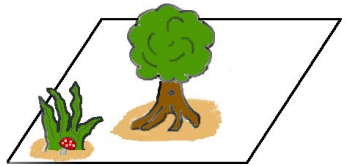
learn

Data



capture

World



[IASI AI]

LIME (Local interpretable model-agnostic explanations)

LIME focuses on training local surrogate models to explain individual predictions.

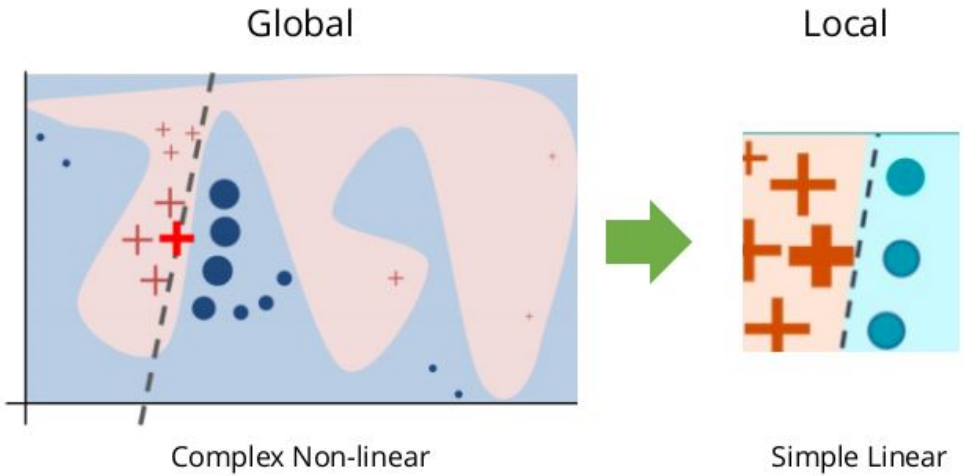
1. For each prediction to explain, permute the observation n times.
2. Let the complex model predict the outcome of all permuted observations.
3. Calculate the distance from all permutations to the original observation.
4. Convert the distance to a similarity score.
5. Select m features best describing the complex model outcome from the permuted data.
6. Fit a simple model to the permuted data, explaining the complex model outcome with the m features from the permuted data weighted by its similarity to the original observation.
7. Extract the feature weights from the simple model and use these as explanations for the complex models local behavior.

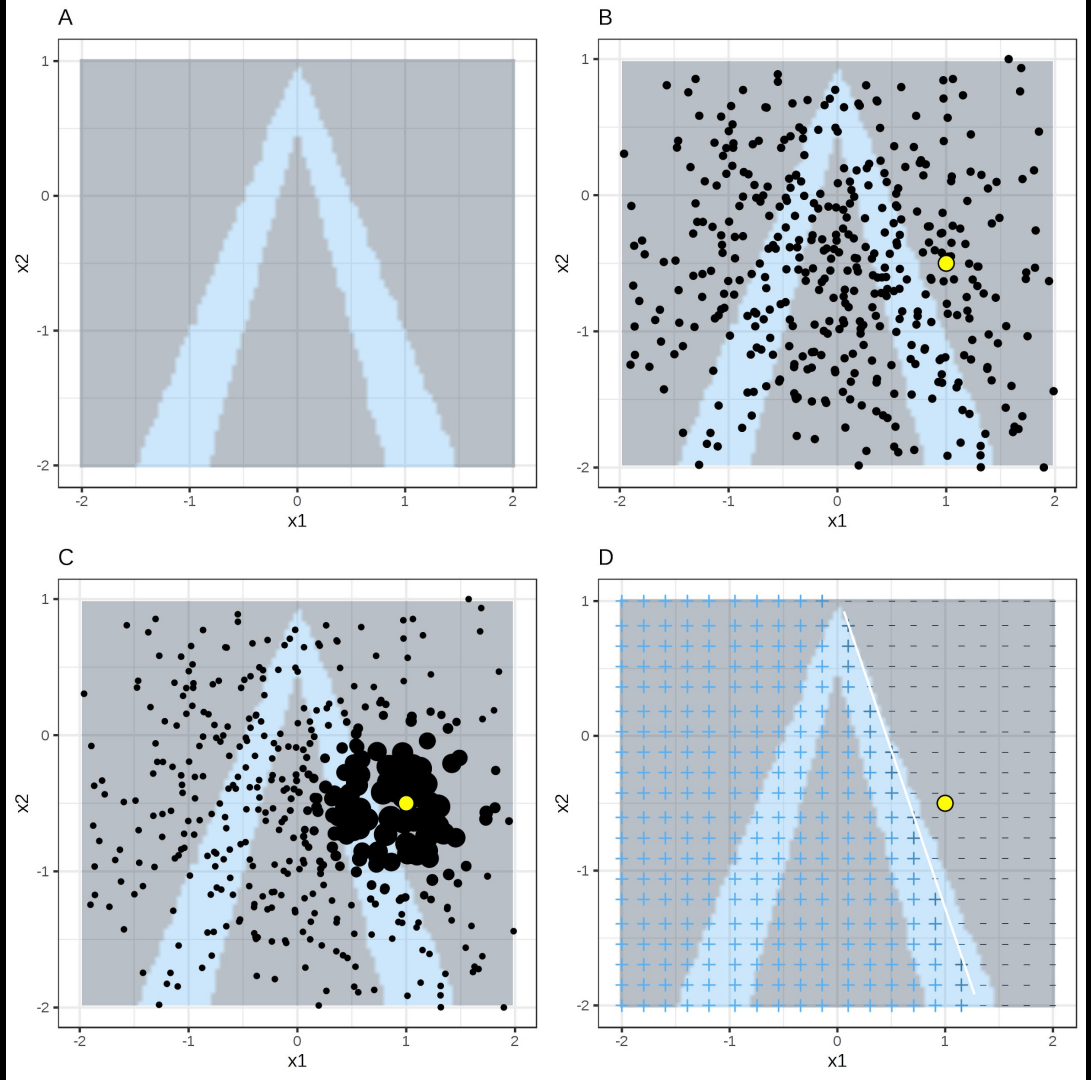
Local versus Global interpretation

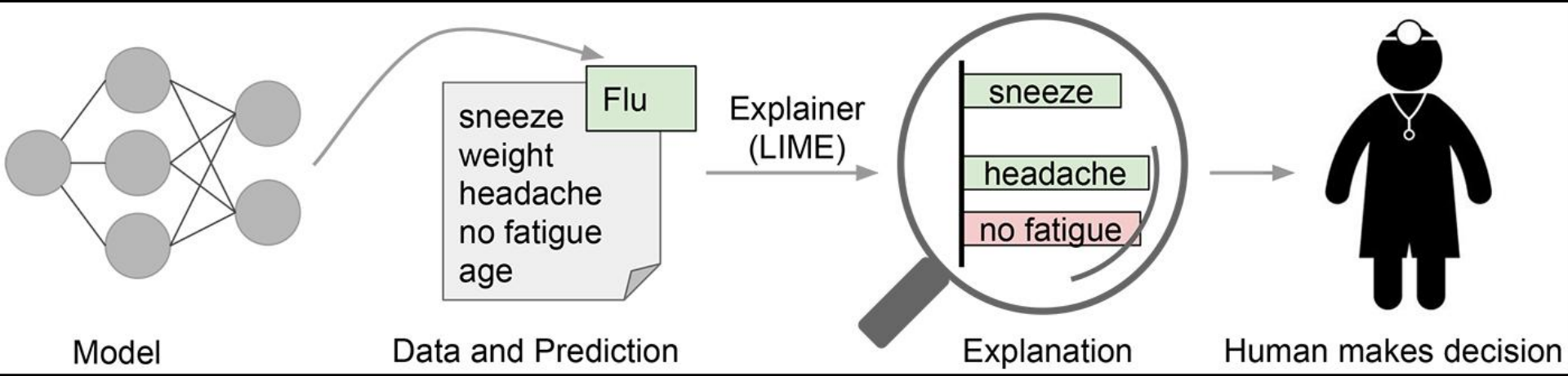
- Globally faithful interpretation might be impossible
- To explain individual decision need to know the small local region

Global trust

- If we trust individual reasonings
- Repeat with a good coverage over the input space
- Local explanation of the + data points
- Locally fitted sparse linear model




A
B
C
D
x1
x2
1
0
-1
-2
2



Explaining individual predictions to a human decision-maker.

- Segments the image using superpixel from opencv

- Build a linear model based on prediction scores against segments

(a) Husky classified as wolf (b) Explanation

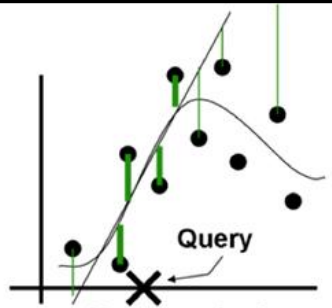
Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: "Husky vs Wolf" experiment results.

Original Image
P(tree frog) = 0.54

Perturbed Instances	P(tree frog)
	0.85
	0.00001
	0.52



Locally weighted regression

Explanation

1. Create a new explainer
   ```
   > my_explainer = Explainer()
   ```
2. Select an observation and create an explanation for it
   ```
   > observation = np.array([...])
   > my_explanation = explainer.explain_instance(observation, predict_function)
   ```
3. Use methods on explanation to visualise results
   ```
   > my_explanation.show_in_notebook()
   > my_explanation.get_image_and_mask()
   > [...]
   ```

SHAP (SHapley Additive exPlanations)

To unify various model explanation methods:

- Model-Agnostic or Model-Specific Approximations
- SHAP provides multiple explainers for different kind of models:
  - *TreeExplainer*: Support XGBoost, LightGBM, CatBoost and scikit-learn models by Tree SHAP.
  - *DeepExplainer* (DEEP SHAP): Support TensorFlow and Keras models by using DeepLIFT and Shapley values.
  - *KernelExplainer* (Kernel SHAP): Applying to any models by using LIME and Shapley values.
  - *GradientExplainer*: Support TensorFlow and Keras models.

SHAP (SHapley Additive exPlanations)

The weight of each feature is computed using Shapley values, method from game theory.

To get the importance of feature X{i}:

- Get all subsets of features S that do not contain X{i}
- Compute the effect on our predictions of adding X{i} to all those subsets

SHAP provides optimisations for different models (linear, trees etc.)

1. Create a new explainer, with our model as argument
   > `explainer = TreeExplainer(my_tree_model)`
2. Calculate shap_values from our model using some observations
   > `shap_values = explainer.shap_values(observations)`
3. Use SHAP visualisation functions with our shap_values
   > `shap.force_plot(base_value, shap_values[0]) # local explanation`
   > `shap.summary_plot(shap_values) # Global features importance`

LIME

- Explanations are short (= selective) and possibly contrastive
- Works for tabular data, text and images
- Implemented in Python and R
- Replace the underlying machine learning model

SHAP

- Contrastive explanations that compare the prediction with the average prediction
- Connects LIME and Shapley values
- Global model interpretations
- Can pinpoint which factors are most impactful for each instance

Disadvantages

LIME

- Correct definition of the neighborhood is an unsolved problem for tabular data.
- Instability of the explanations
- Data points are sampled from the initial distribution of the feature, ignoring the correlation between features

SHAP

- KernelSHAP is slow
- KernelSHAP ignores feature dependence
- Access to data is needed to compute them for new data (except for TreeSHAP)

[IASI AI]

Demo

Bibliography

Thanks for the ideas:

- https://arxiv.org/pdf/1602.04938.pdf
- https://arxiv.org/pdf/1702.08608.pdf
- https://christophm.github.io/interpretable-ml-book
- https://cran.r-project.org/web/packages/lime/vignettes/Understanding_lime.html
- https://www.youtube.com/watch?v=KP7-JtFMLo4
- https://www.youtube.com/watch?v=d4PPMpdUCz8
- https://www.youtube.com/watch?v=d4PPMpdUCz8
- https://www.youtube.com/watch?v=Q8rTrmqUQsU
- https://towardsdatascience.com/explain-any-models-with-the-shap-values-use-the-kernelexplainer-79de9464897a
- https://towardsdatascience.com/explain-your-model-with-the-shap-values-bc36aac4de3d
- https://medium.com/intel-student-ambassadors/local-interpretable-model-agnostic-explanations-lime-the-eli5-way-b4fd61363a5e
- https://www.oreilly.com/radar/ideas-on-interpreting-machine-learning/
- https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime
- https://www.slideshare.net/StampedeCon/interpretability-why-should-we-trust-youinterpretability-of-deep-neural-networks
- https://blog.dominodatalab.com/shap-lime-python-libraries-part-1-great-explainers-pros-cons/

Thanks for the code:

- https://github.com/slundberg/shap
- https://github.com/marcotcr/lime
- https://www.youtube.com/watch?v=C80SQe16Rao

[IASI AI]

Thank You!

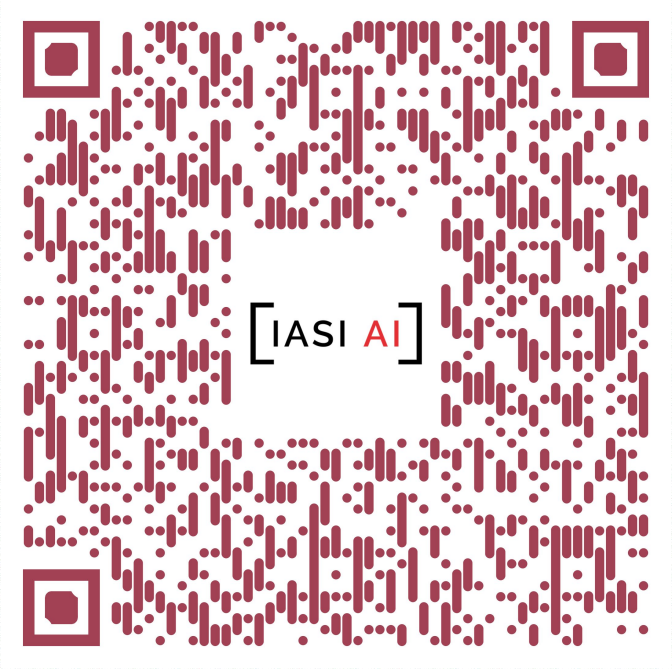
iasi.ai

fb.me/AI.Iasi/

meetup.com/IASI-AI/

[IASI AI]

We ❤ Feedback



[IASI AI]

Community partners