Explainable AI (XAI)

Daisuke Sato

2020/07/11

Reference (Survey on XAI)

- A. Adadi, M. Berrada, https://ieeexplore.ieee.org/document/8466590/
- Finale Doshi-Velez, Been Kim, arXiv:1702.08608

Codes

Notebook in Kaggle by Daisuke Sato

- “Random forest/XGBoost, and LIME/SHAP with Titanic”
- “Predict Housing price”

Self-Introduction

- Daisuke Sato, Ph.D.
  - Graduate school
    - Kyoto university
    - Research topic: Theoretical Physics (Quark-Gluon Plasma)
  - Previous jobs
    - Postdoc (US., Italy, Germany)
  - Current position
    - CTO

Contents

- **Concept/Motivation**
- **Recent trends on XAI**
- **Method 1: LIME/SHAP**
  - Example: Classification
  - Example: Regression
  - Example: Image classification
- **Method 2: ABN for image classification**

Concept/Motivation

Generally speaking, AI is a blackbox.

We want AI to be explainable because…

1. **Users should trust AI to actually use it (prediction itself, or model)**

   Ex: diagnosis/medical check, credit screening

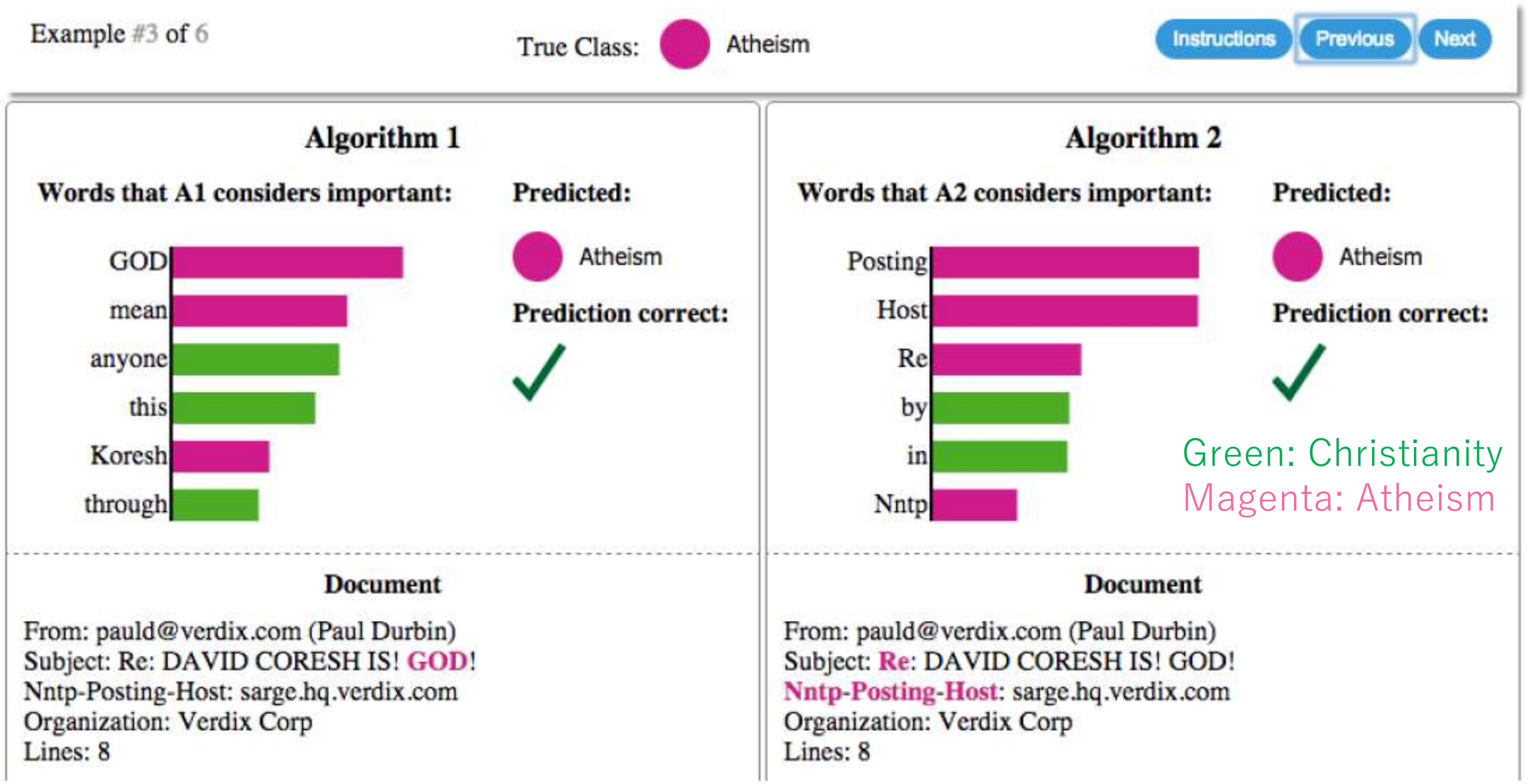
G. Tolomei, et. al., arXiv:1706.06691

People want to know why they were rejected by AI screening,
and what they should do in order to pass the screening.

Concept/Motivation

2. It helps to choose a model from some candidates

Classifier of text to “Christianity” or “Atheism” (無神論)



Example #3 of 6

True Class: ● Atheism

Algorithm 1

Words that A1 considers important:

Predicted: ● Atheism

Prediction correct: ✓

Document

From: pauld@verdix.com (Paul Durbin)
Subject: Re: DAVID CORESH IS! **GOD**!
Nntp-Posting-Host: sarge.hq.verdix.com
Organization: Verdix Corp
Lines: 8

Algorithm 2

Words that A2 considers important:

Predicted: ● Atheism

Prediction correct: ✓

Green: Christianity
Magenta: Atheism

Document

From: pauld@verdix.com (Paul Durbin)
Subject: **Re**: DAVID CORESH IS! GOD!
Nntp-Posting-Host: sarge.hq.verdix.com
Organization: Verdix Corp
Lines: 8

Both model give correct classification,
but it is apparent that model 1 is better than model 2.

Concept/Motivation

3. **It is useful to find overfitting, when train data is different from test data**

Cf: Famous example of “husky or wolf”

Training dataset contains pictures of wolfs with snowy background.

(a) Husky classified as wolf

Then, the classifier trained on that dataset outputs “wolf” if the input image contains snow.

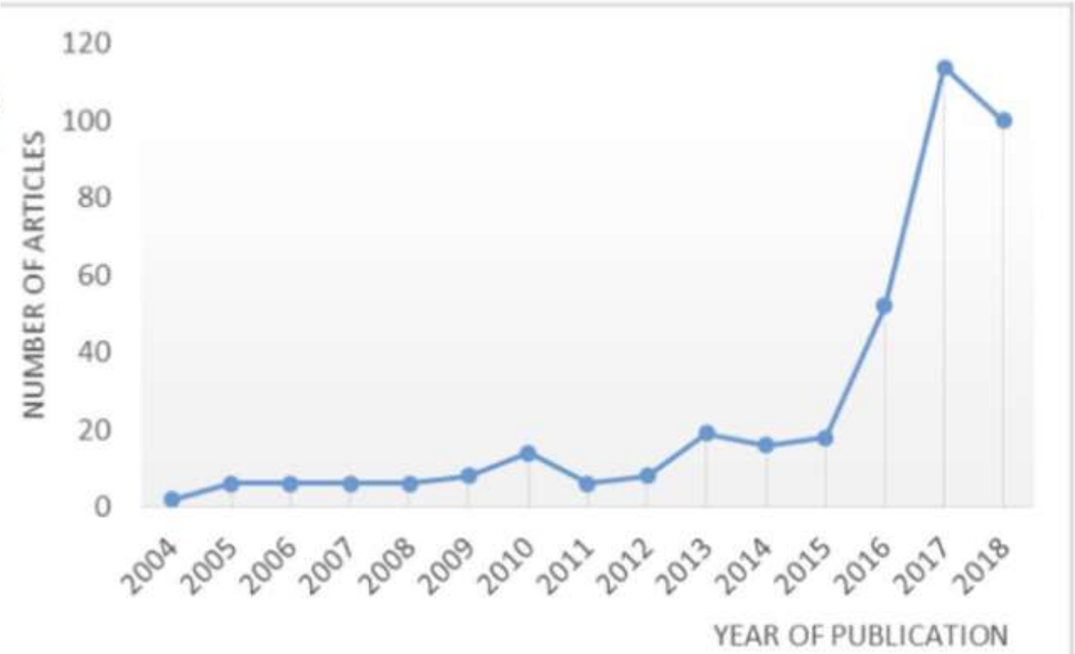
Contents

- Concept/Motivation
- **Recent trends on XAI**
- Method 1: LIME/SHAP
  - Example: Classification
  - Example: Regression
  - Example: Image classification
- Method 2: ABN for image classification

Arithmer

Recent trends

# of papers which includes
one of explanation-related words (“intelligible”, “interpretable”,…)
AND
one of AI-related words (“Machine learning”, “deep learning”,…)
FROM
7 repositories (arXiv, Google scholar, …)



Recently, researchers are studying XAI more and more.

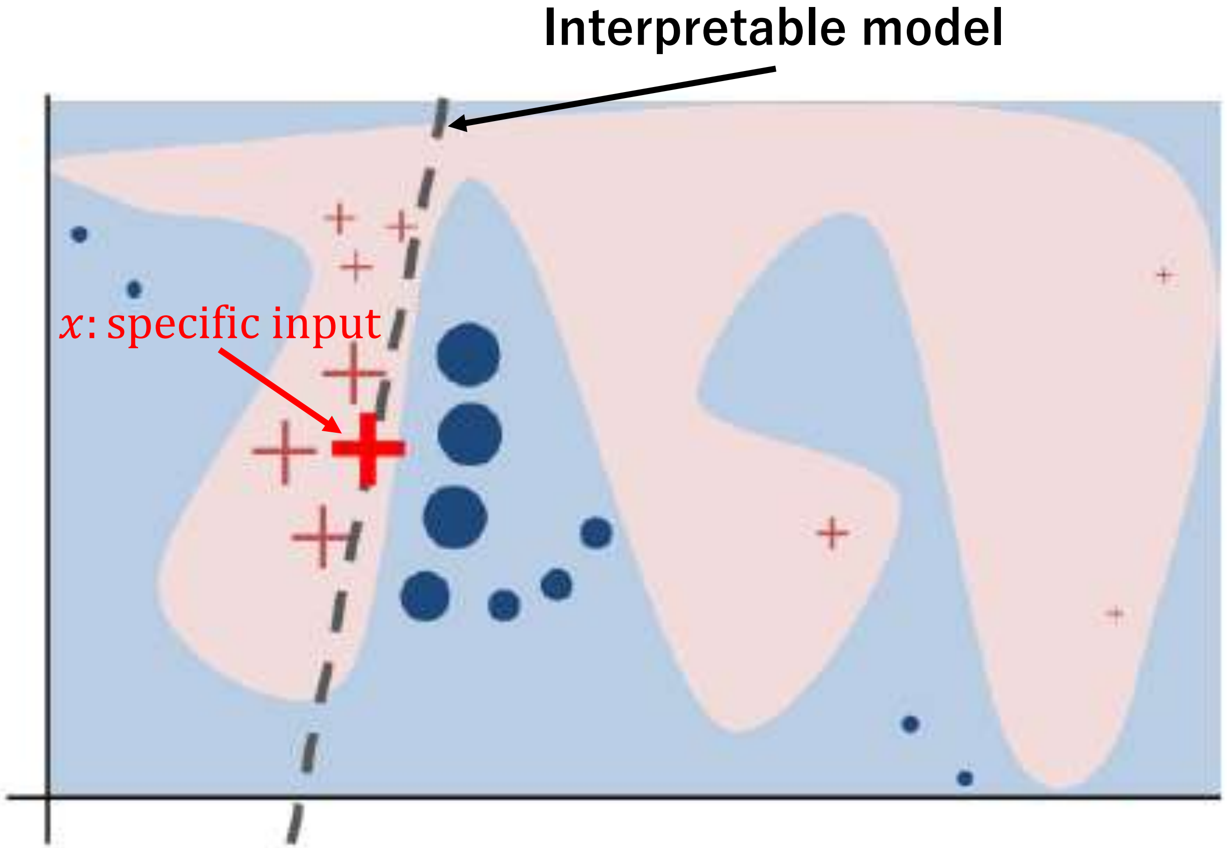
Contents

- Concept/Motivation
- Recent trends on XAI
- **Method 1: LIME/SHAP**
  - Example: Classification
  - Example: Regression
  - Example: Image classification
- Method 2: ABN for image classification

Method 1: Local Interpretable Model-agnostic Explanations (LIME)

M. Ribeiro, S. Singh, C. Guestrin, arXiv:1602.04938

Objects: Classifier or Regressor



Basic idea: Approximate original ML model with interpretable model (linear model/DT), in the vicinity of specific features.

Method 1: Local Interpretable Model-agnostic Explanations (LIME)

Original model

$$\begin{tikzcd}
\mathbb{R}^d \arrow[r, "f"] & \mathbb{R} \\
\rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} \\
x & f(x)
\end{tikzcd}$$

features *prediction*

Interpretable model

$$\begin{tikzcd}
\{0,1\}^{d'} \arrow[r, "g"] & \mathbb{R} \\
\rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} \\
z' & g(z')
\end{tikzcd}$$

Absence/presenc e of features *prediction*

Distance of predictions of original model (f) and its approximation

Complexity of model (depth of threes for DT/ # of non-zero weights for linear model)

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

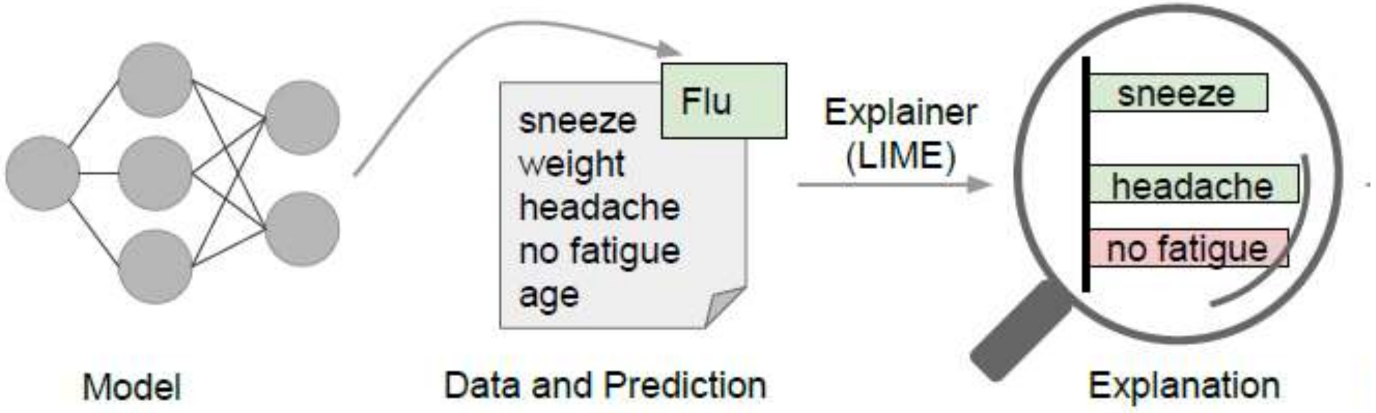
$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathcal{Z}} \pi_x(z) \left(f(z) - g(z')\right)^2$$

Linear model

$$g(z') = w_g \cdot z'$$

Weights of sampling $\pi_x(z) = \exp(-D(x, x')^2/\sigma^2)$

Explaining individual explanation



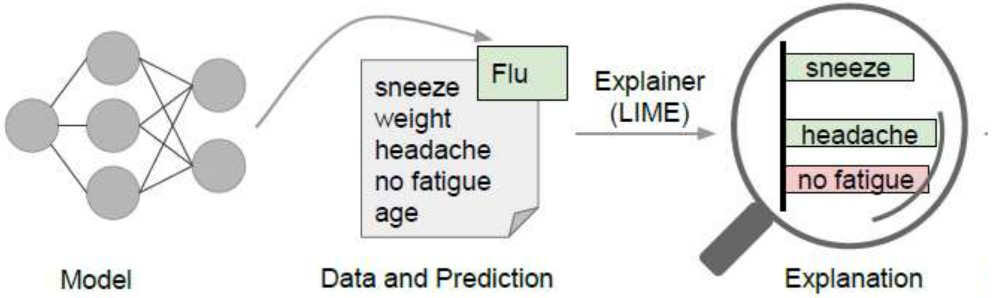
1. Original model predicts from features (sneeze, headache,···) whether the patient is flu or not.

2. LIME approximates the model with linear model in the vicinity of the specific patient.

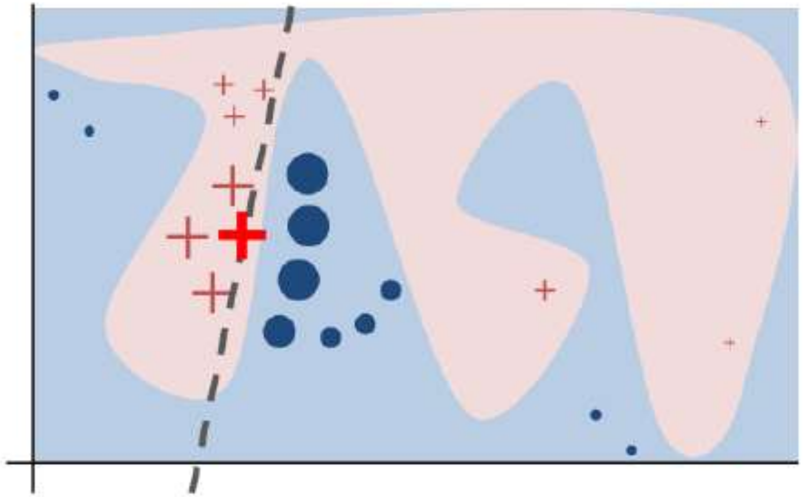
3. The weights of the linear model for each feature give “explanation” of the prediction

desirable features of LIME

- **Interpretable**



- **Local fidelity**
- **Model-agnostic**
  (Original model is not affected by LIME at all)
- **Global perspective**
  (sample different inputs and its predictions)



Shapley Additive exPlanations (SHAP)

S. Lundberg, S-I. Lee, arXiv:1705.07874

Generalization of methods for XAI,

- LIME
- DeepLIFT A. Shrikumar et. al., arXiv:1605.01713
- Layer-Wise Relevance Propagation Sebastian Bach et al. In: PloS One 10.7 (2015), e0130140

Actually, they are utilizing concepts of cooperative game theory:

- Shapley regression values
- Shapley sampling values
- Quantitative Input Influence

Example: Classification (Titanic dataset)

Label

Features

All values are unique

77% are NaN

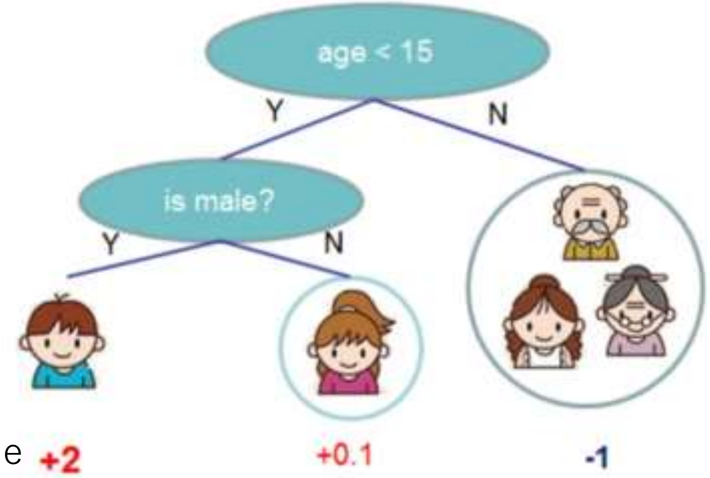
Variable	Definition	Key
survival	Survival	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
~~ticket~~	~~Ticket number~~	
fare	Passenger fare	
~~cabin~~	~~Cabin number~~	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

# of training samples = 891 →Train : Validation = 8 : 2
# of test samples = 418

GBDT

Decision Tree (DT)

Selection of features and its threshold is done, so that the mutual information is maximized.



“Boyness” value

Entropy in parent node

Entropy in child nodes

Mutual information: $I_H(D_p) - I_H(D_{left}) - I_H(D_{right})$

Entropy at a node is maximized when the samples in that node are uniformly distributed to all the classes.

→**Child nodes tends to be purely distributed**

☺

- Simple to understand and interpret
- Can represent interaction among features unlike linear model (at least up to # of depth_tree)
- No need to NaN preprocessing/standardization

☹

- Tends to overlearn (if trees are deep)

GBDT

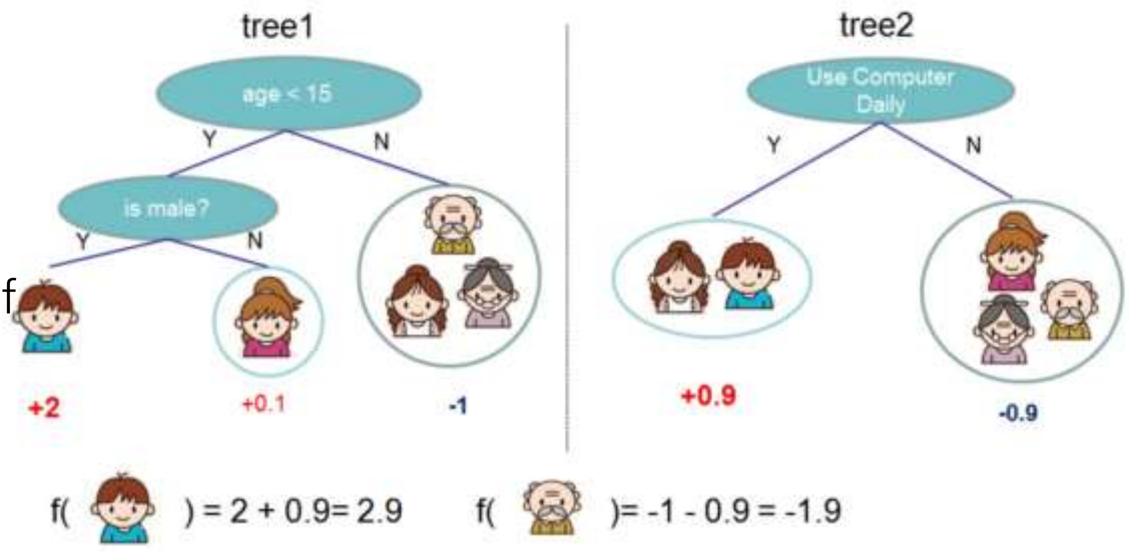
Gradient Boosting Decision Tree (GBDT)

T. Chen, C. Guestrin, arXiv:1603.02754

In ensemble method, the final value is calculated from average of many trees.

How to decide the structure of tree ensemble (forest):

1. Start with one tree
2. Add another tree so that the loss function is minimized (gradient boost)



Cf: In random forest, all the trees are generated in parallel, at once.
Compared with it, GBDT can learn from the misprediction in the previous tree, so the result is expected to be better.

☺

- Overlearning problem is improved

☹

- Less interpretability
- Many hyperparameters to be tuned

Very popular method in Kaggle

GBDT

I used one of GBDT (XGBoost) as a ML model, under the following conditions:

- No standardization of numerical features (as DT does not need it)
- No postprocessing of NaN (GBDT treats NaN as it is)
- No feature engineering
- Hyperparameter tuning for n_estimators and max_depth (optuna is excellent)

Results:

Best parameters: {'n_estimators': 20, 'max_depth': 12}
Validation score: 0.8659217877094972

Test score: 0.77033

Cf: Baseline (all women survive, all men die): 0.76555

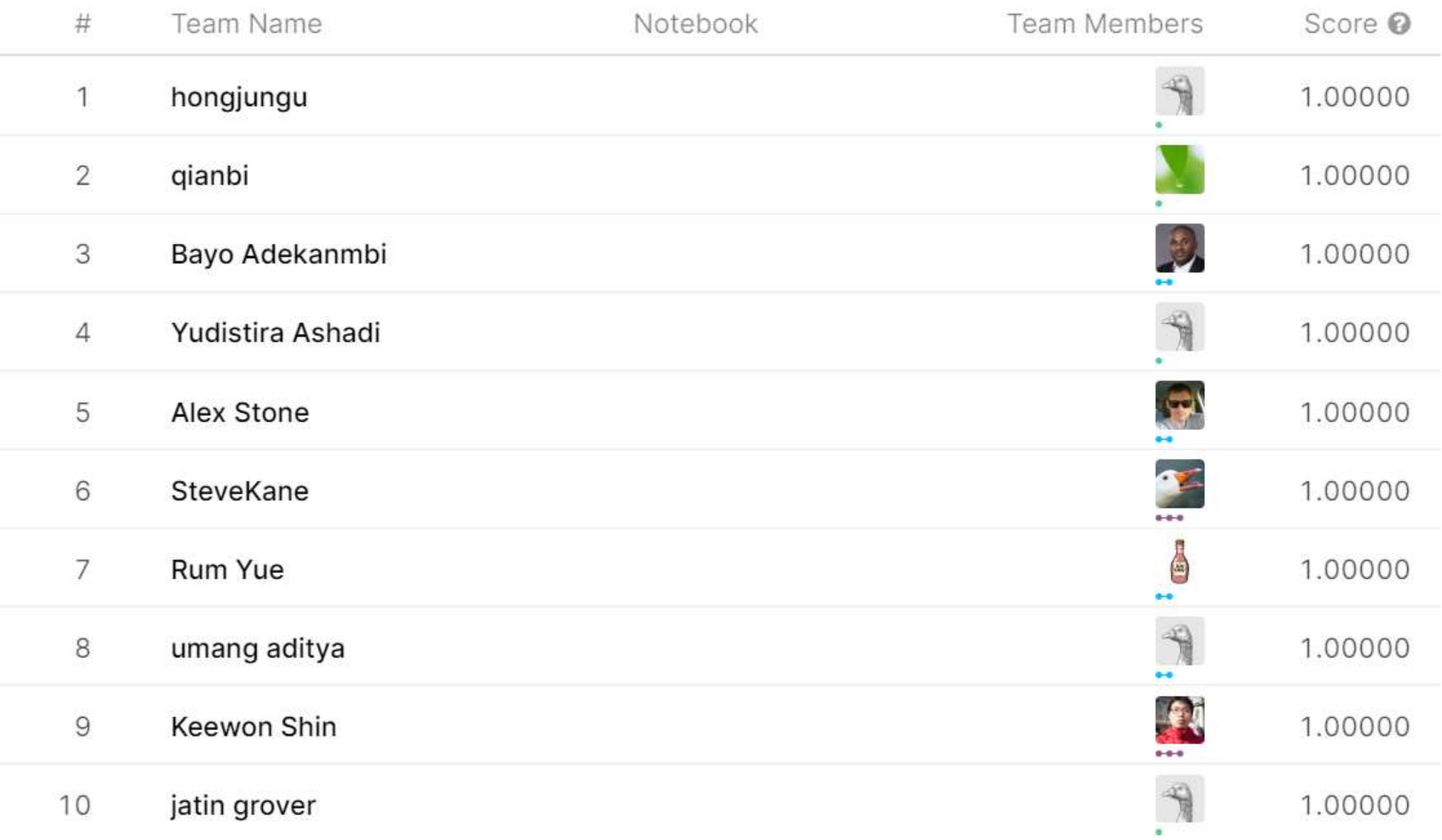
Not satisfying…

Cf: Reported best score with ensemble method: 0.84210

Review of know-how on feature engineering by experts:
Kaggle notebook “How am I doing with my score?”

GBDT

Cf: Test score at Kaggle competition



#	Team Name	Notebook	Team Members	Score
1	hongjungu			1.00000
2	qianbi			1.00000
3	Bayo Adekanmbi			1.00000
4	Yudistira Ashadi			1.00000
5	Alex Stone			1.00000
6	SteveKane			1.00000
7	Rum Yue			1.00000
8	umang aditya			1.00000
9	Keewon Shin			1.00000
10	jatin grover			1.00000

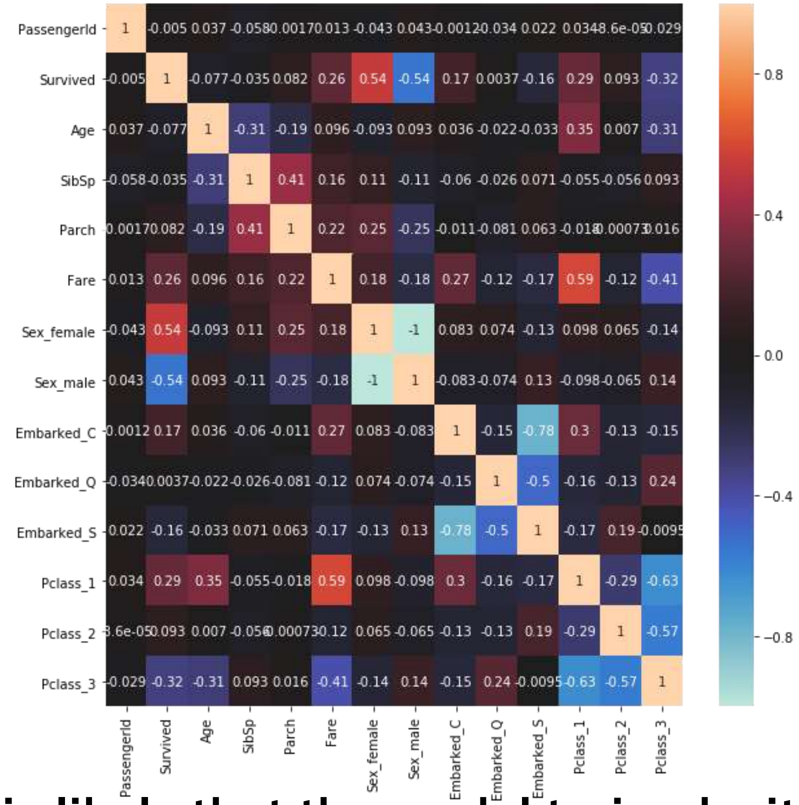
Cheaters… (Memorizing all the names of the survivors? Repeat trying with GA?)

GBDT

Linear correlation between features and label

（explanation of data itself, not model)

The method which you try first, to select important features.
You can work when the number of features is small.



	PassengerId	Survived	Age	SibSp	Parch	Fare	Sex_female	Sex_male	Embarked_C	Embarked_Q	Embarked_S	Pclass_1	Pclass_2	Pclass_3
PassengerId	1	-0.005	0.037	-0.058	0.0017	0.013	-0.043	0.043	-0.0012	-0.034	0.022	0.034	8.6e-05	0.029
Survived	-0.005	1	-0.077	-0.035	0.082	0.26	0.54	-0.54	0.17	0.0037	-0.16	0.29	0.093	-0.32
Age	0.037	-0.077	1	-0.31	-0.19	0.096	-0.093	0.093	0.036	-0.022	-0.033	0.35	0.007	-0.31
SibSp	-0.058	-0.035	-0.31	1	0.41	0.16	0.11	-0.11	-0.06	-0.026	0.071	-0.055	-0.056	0.093
Parch	0.0017	0.082	-0.19	0.41	1	0.22	0.25	-0.25	-0.011	-0.081	0.063	-0.018	0.00073	0.016
Fare	0.013	0.26	0.096	0.16	0.22	1	0.18	-0.18	0.27	-0.12	-0.17	0.59	-0.12	-0.41
Sex_female	-0.043	0.54	-0.093	0.11	0.25	0.18	1	-1	0.083	0.074	-0.13	0.098	0.065	-0.14
Sex_male	0.043	-0.54	0.093	-0.11	-0.25	-0.18	-1	1	-0.083	-0.074	0.13	-0.098	-0.065	0.14
Embarked_C	-0.0012	0.17	0.036	-0.06	-0.011	0.27	0.083	-0.083	1	-0.15	-0.78	0.3	-0.13	-0.15
Embarked_Q	-0.034	0.0037	-0.022	-0.026	-0.081	-0.12	0.074	-0.074	-0.15	1	-0.5	-0.16	-0.13	0.24
Embarked_S	0.022	-0.16	-0.033	0.071	0.063	-0.17	-0.13	0.13	-0.78	-0.5	1	-0.17	0.19	-0.0095
Pclass_1	0.034	0.29	0.35	-0.055	-0.018	0.59	0.098	-0.098	0.3	-0.16	-0.17	1	-0.29	-0.63
Pclass_2	8.6e-05	0.093	0.007	-0.056	0.00073	-0.12	0.065	-0.065	-0.13	-0.13	0.19	-0.29	1	-0.57
Pclass_3	-0.029	-0.32	-0.31	0.093	0.016	-0.41	-0.14	0.14	-0.15	0.24	-0.0095	-0.63	-0.57	1

Correlation between Survived and features

Sex_female	**0.543351**
Pclass_3	**0.322308**
Pclass_1	0.285904
Fare	0.257307
Embarked_C	0.168240
Embarked_S	0.155660

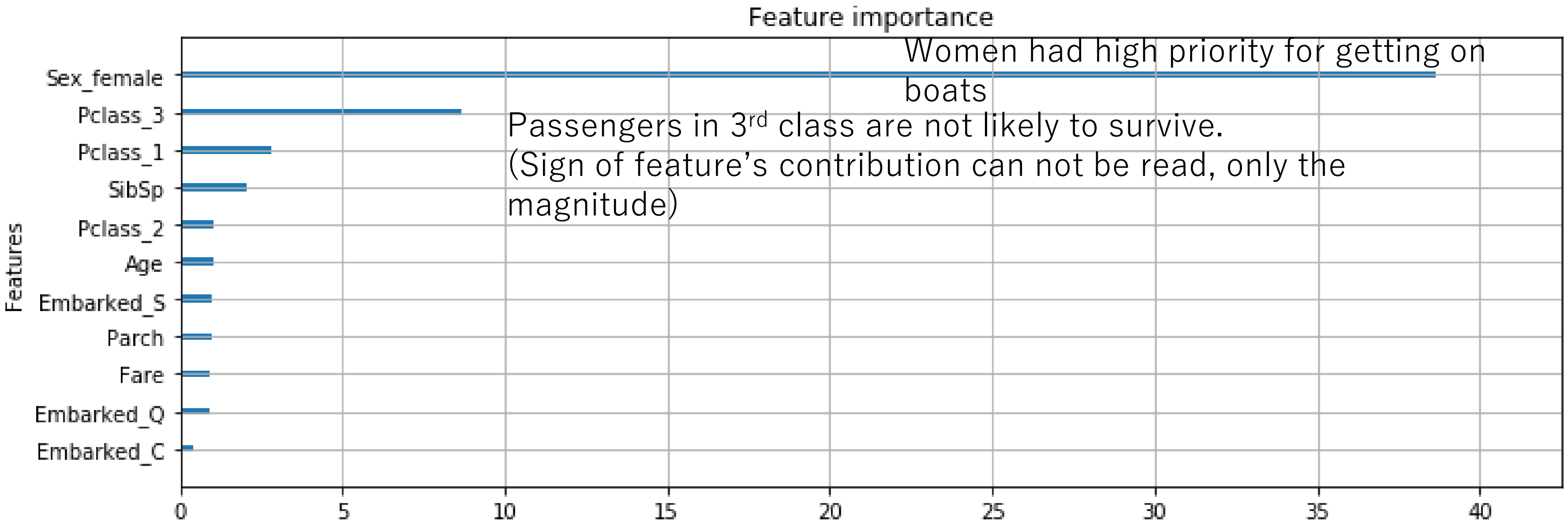
Fare has large correlation with Pclass, as you can easily expect (~0.4-0.6).
Also with Embarked_C (~0.27), which I do not know why.

It is likely that the model trained with this data put large importance on the features with large correlation.

GBDT

Importance of features in GBDT

(explanation of whole model, not for specific sample)



Women had high priority for getting on boats

Passengers in 3rd class are not likely to survive.
(Sign of feature's contribution can not be read, only the magnitude)

The average gain (for mutual information, or inpurity) of splits which use the feature

Top 3 agrees with that in linear correlation.

GBDT

However, this method has a problem of “*inconsistency*” (when a model is changed such that a feature has a higher impact on the output, the importance of that feature can decrease)

This problem is overcome by LIME/SHAP.

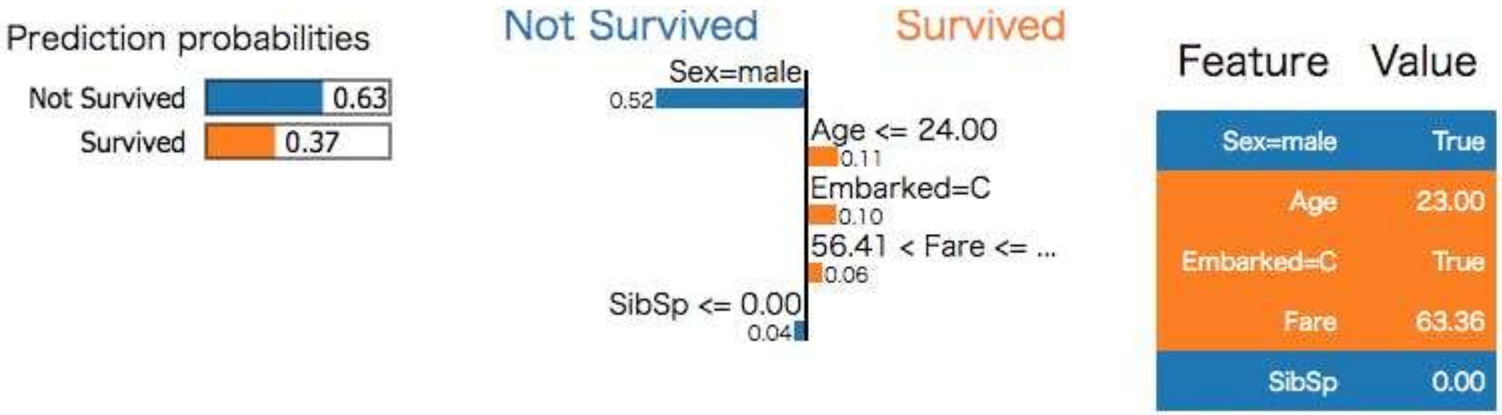
LIME

Explanation by LIME

(explanation of model prediction for specific sample)

My code did not work on Kaggle kernel, because of a bug in LIME package…
So, here I quote results from other person.

https://qiita.com/fufufukakaka/items/d0081cd38251d22ffebf



Feature	Value
Sex=male	True
Age	23.00
Embarked=C	True
Fare	63.36
SibSp	0.00

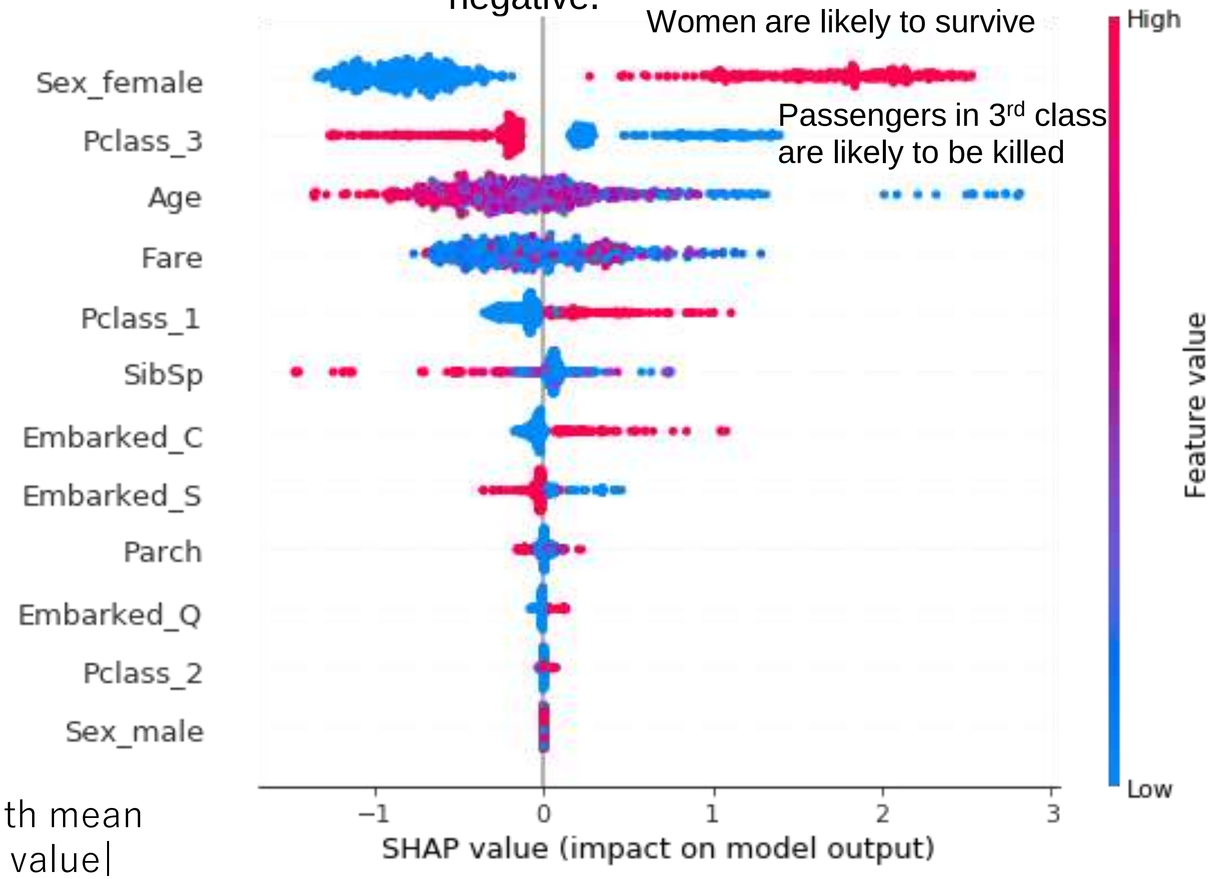
As LIME approximates the model with linear function locally, the weights of the features are different depending on sample.

In this sample, the top 3 features are Sex, Age, and Embarked.

SHAP

Results of SHAP

Red points（Sex_female=1:female） have positive SHAP values, while the blue points （Sex_ｆemale=0:male） negative.



Women are likely to survive

Passengers in 3rd class are likely to be killed

Sorted with mean of |SHAP value|

Top 3 does not agree with that in linear correlation/XGBoost score (Age enters).

SHAP is consistent (unlike feature importance of XGBoost) and has local fidelity (unlike linear correlation), I would trust SHAP result than the other two.

Example: Regression（Ames Housing dataset）

Label

Features

Variable	Definition	Key
SalePrice	the property's sale price in dollars	-
OverallQual	Rates the overall material and finish of the house	10:Very Excellent 9:Excellent … 1:Very Poor
GarageCars	Size of garage in car capacity	-
KitchenAbvGr	Kitchens above grade	Ex:Excellent Gd:Good TA:Typical/Average Fa:Fair Po:Poor
…(80 features in total)	…	…

Dataset describing the sale of individual residential property in Ames, Iowa, from 2006 to 2010.

# of training samples = 1460 →Train : Validation =75 : 25

# of test samples =1459

Example: Regression (Ames Housing dataset)

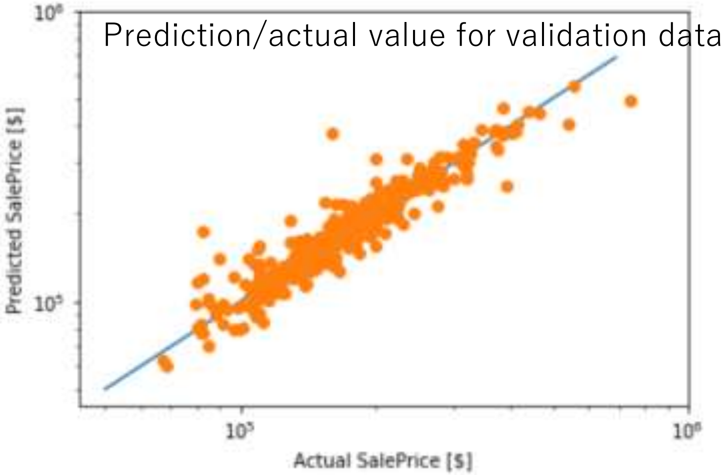
Model: XGBoost

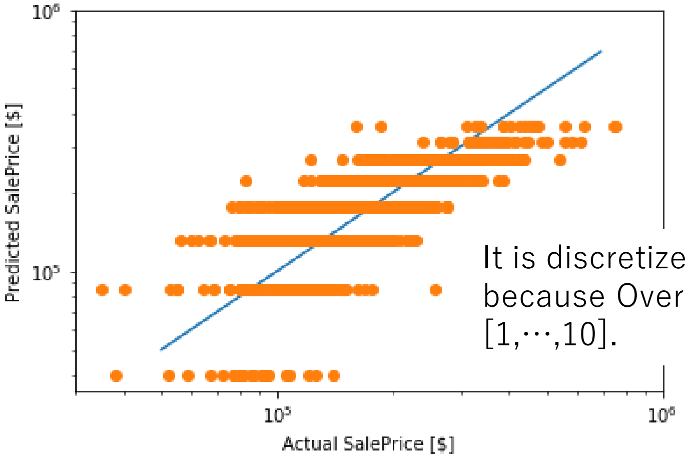
Results (metric is Root Mean Squared Error of Log (RMSEL))

Validation : 0.13830

Test: 0.16332

Cf: Baseline (linear regression in terms of OverallQual): 1.1608





It is discretized because OverallQual is [1,⋯,10].

This time, it is much better than simple baseline.

Cf: Score with the same method (XGBoost), but with feature engineering: 0.11420

Kaggle notebook “Stacked Regressions : Top 4% on LeaderBoard”

Example: Regression (Ames Housing dataset)

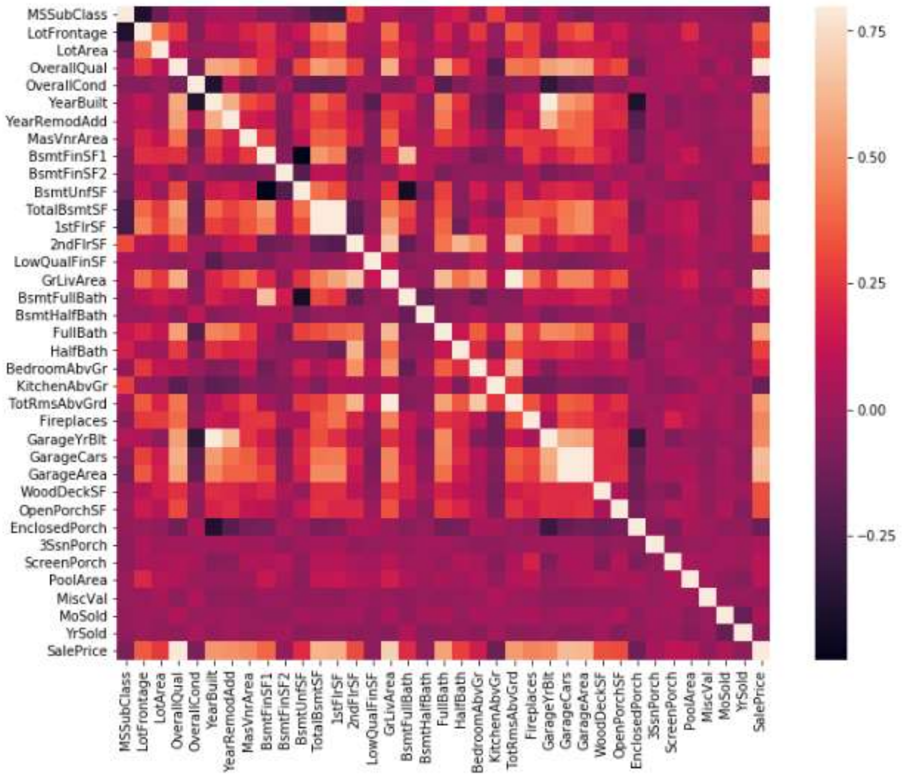
Linear correlation between features and label

(explanation of data itself, not model)

Rates the overall material and finish of the house

Above grade (ground) living area square feet

Size of garage in car capacity



Correlation between SalePrice and features

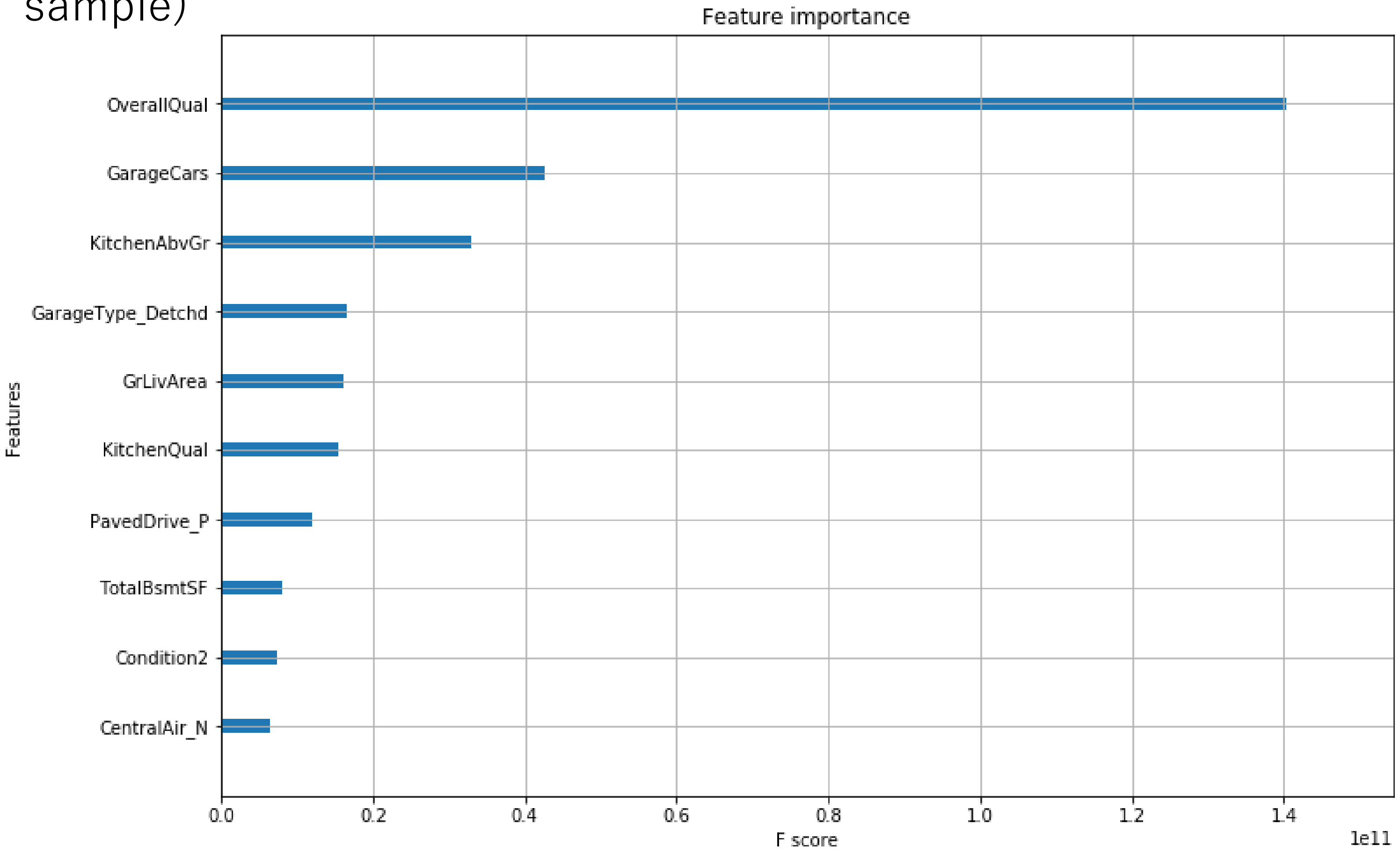
OverallQual	**0.790982**
GrLivArea	**0.708624**
GarageCars	**0.640409**
GarageArea	0.623431
TotalBsmtSF	0.613581
1stFlrSF	0.605852
FullBath	0.560664
TotRmsAbvGrd	0.533723
YearBuilt	0.522897
YearRemodAdd	0.50710

…

Example: Regression (Ames Housing dataset)

Importance of features in GBDT

(explanation of whole model, not for specific sample)



Top 3 is different from linear correlation (KitchenAbvGr).

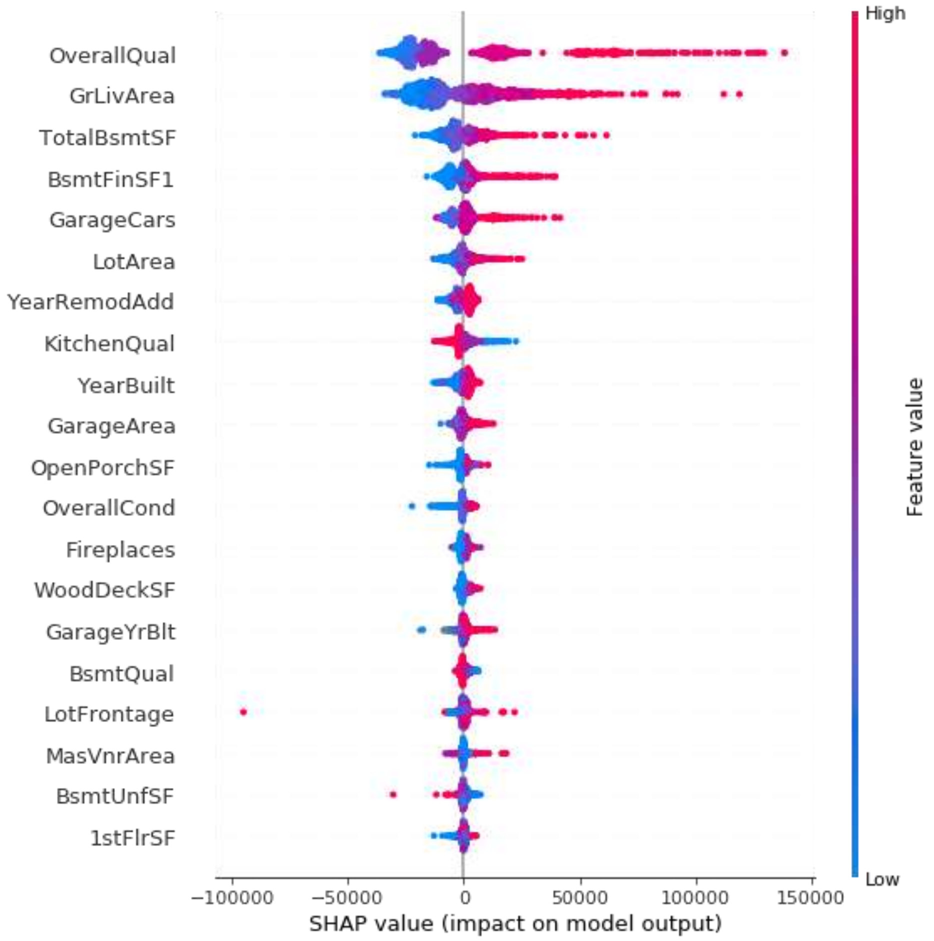
Example: Regression (Ames Housing dataset)

Explanation by SHAP

(explanation of model prediction for specific sample)

GrLivArea: Above grade (ground) living area square feet

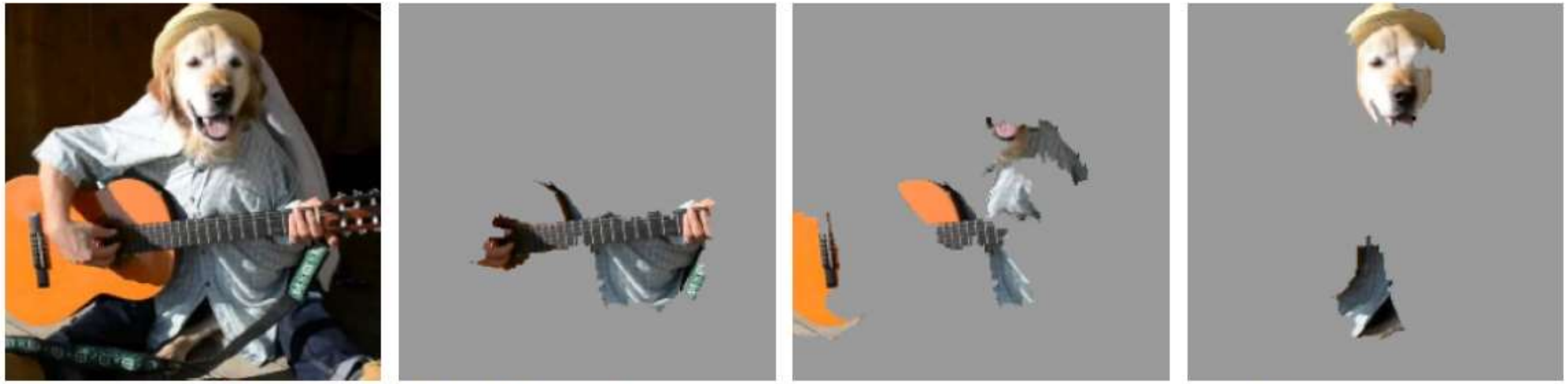
TotalBsmtSF: Total square feet of basement area



Top 3 is different from linear correlation/LIME.

Example: Image Classification

Results of LIME



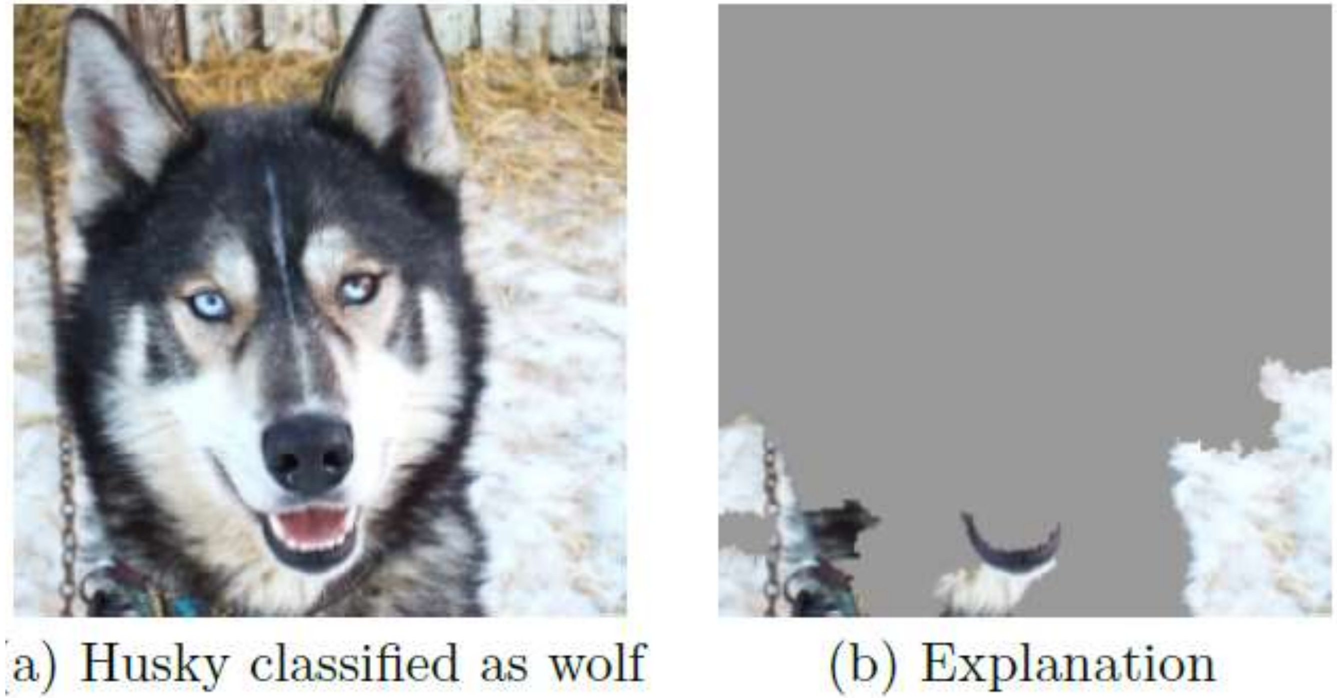
(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

The model seems to focus on the right places.

However, there are models which can not be approximated with LIME.
Ex: Classifier whether the image is “retro” or not considering the
values of the whole pixels (sepia?)

Example: Image Classification

Husky or wolf example



(a) Husky classified as wolf

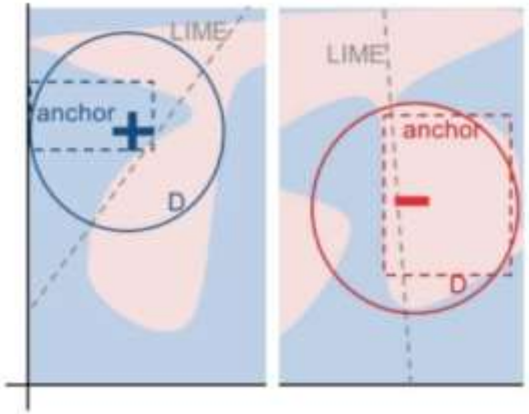
(b) Explanation

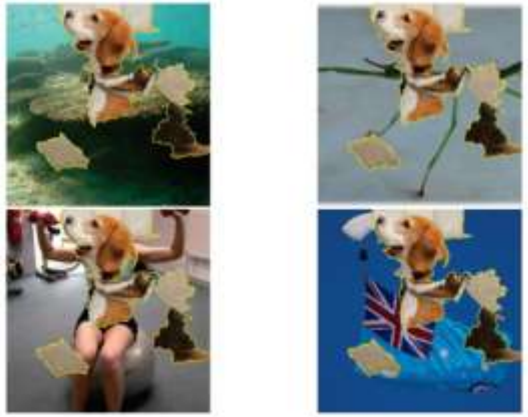
By looking at this explanation,
it is easy to find that the model is focusing on snow.

Example: Image Classification

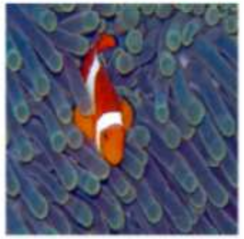
Other approaches

- Anchor M. Ribeiro, et. al., *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.
  Gives range of features which does not change the prediction




(c) Images where Inception predicts $P(\text{beagle}) > 90\%$

- Influence P. Koh, P. Liang, arXiv:1703.04730
  Gives **training data** which the prediction is based on


Test image

Training images which affect the prediction of the test image

Contents

- Concept/Motivation
- Recent trends on XAI
- Method 1: LIME/SHAP
  - Example: Classification
  - Example: Regression
  - Example: Image classification
- **Method 2: ABN for image classification**

Method 2: ABN for image classification

Class Activation Mapping (CAM)

B. Zhou, et. al., arXiv:1512.04150

x, y: spatial coordinate
k: channel

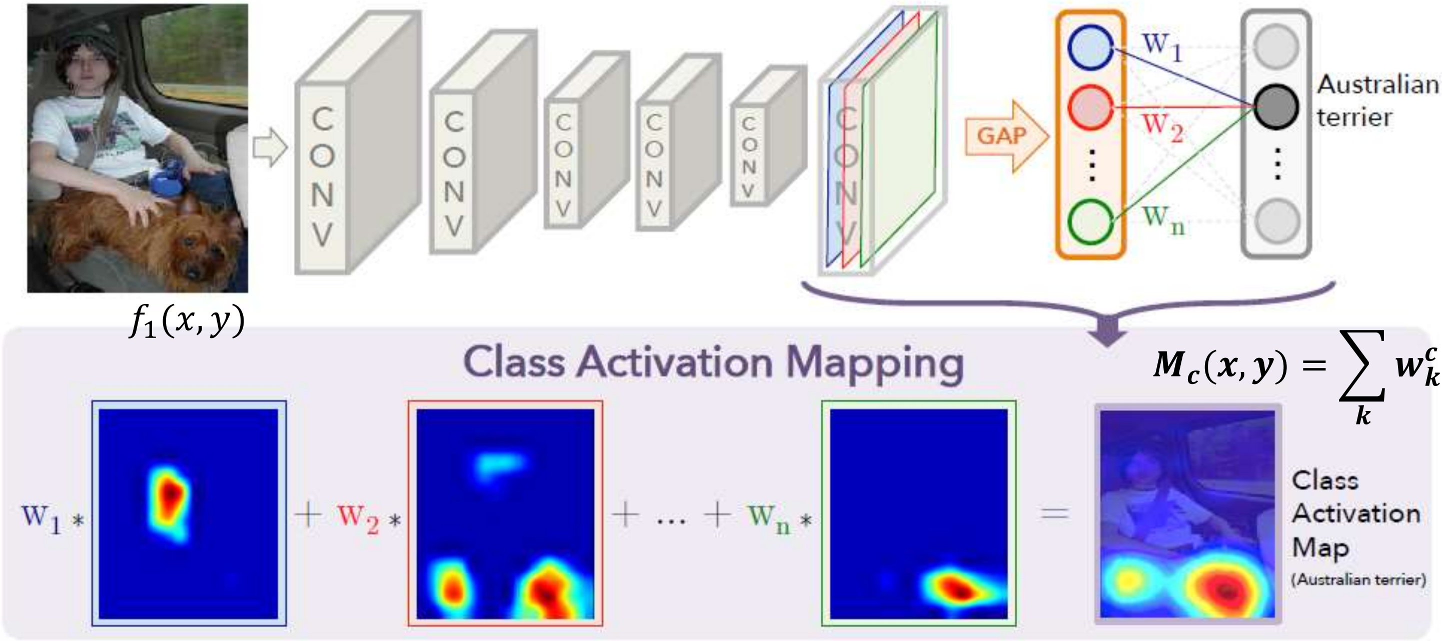
Global Average Pooling (GAP) for channel c

$$F_k = \sum_{x,y} f_k(x,y)$$

Score for class c

$$S_c = \sum_k w_k^c F_k = \sum_{x,y} M_c(x,y)$$

$f_k(x,y)$



$f_1(x,y)$

$$\boldsymbol{M_c(x,y) = \sum_k w_k^c f_k(x,y)}$$

☹

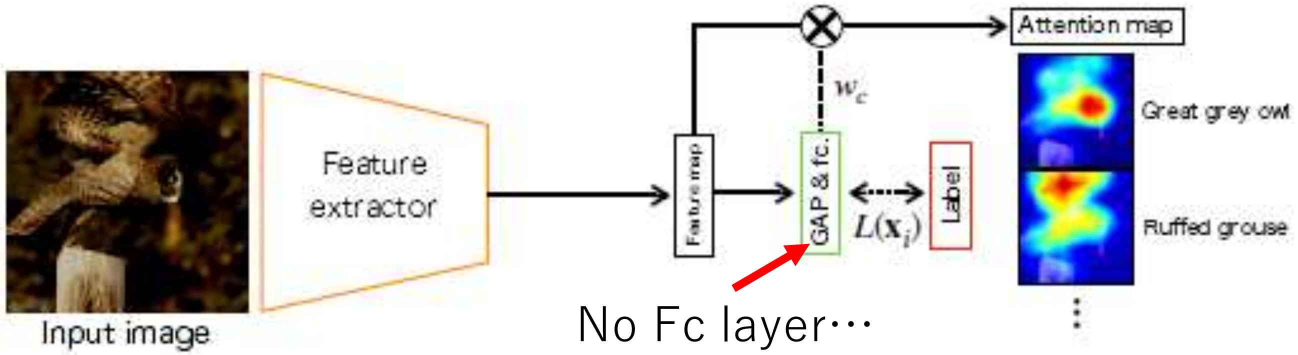
- Decrease classification accuracy because fully-connected (Fc) layer is replaced with GAP.

Method 2: ABN for image classification

Attention Branch Network (ABN)

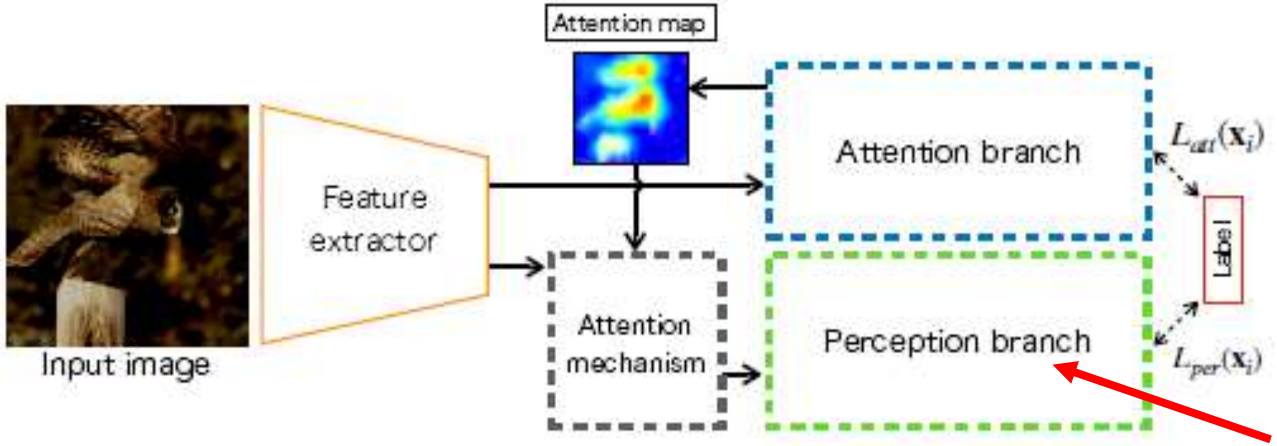
H. Fukui, et. al., arXiv:1812.10025

Basic idea: Divide attention branch from classification branch so that Fc layers can be used in the latter branch.



No Fc layer…

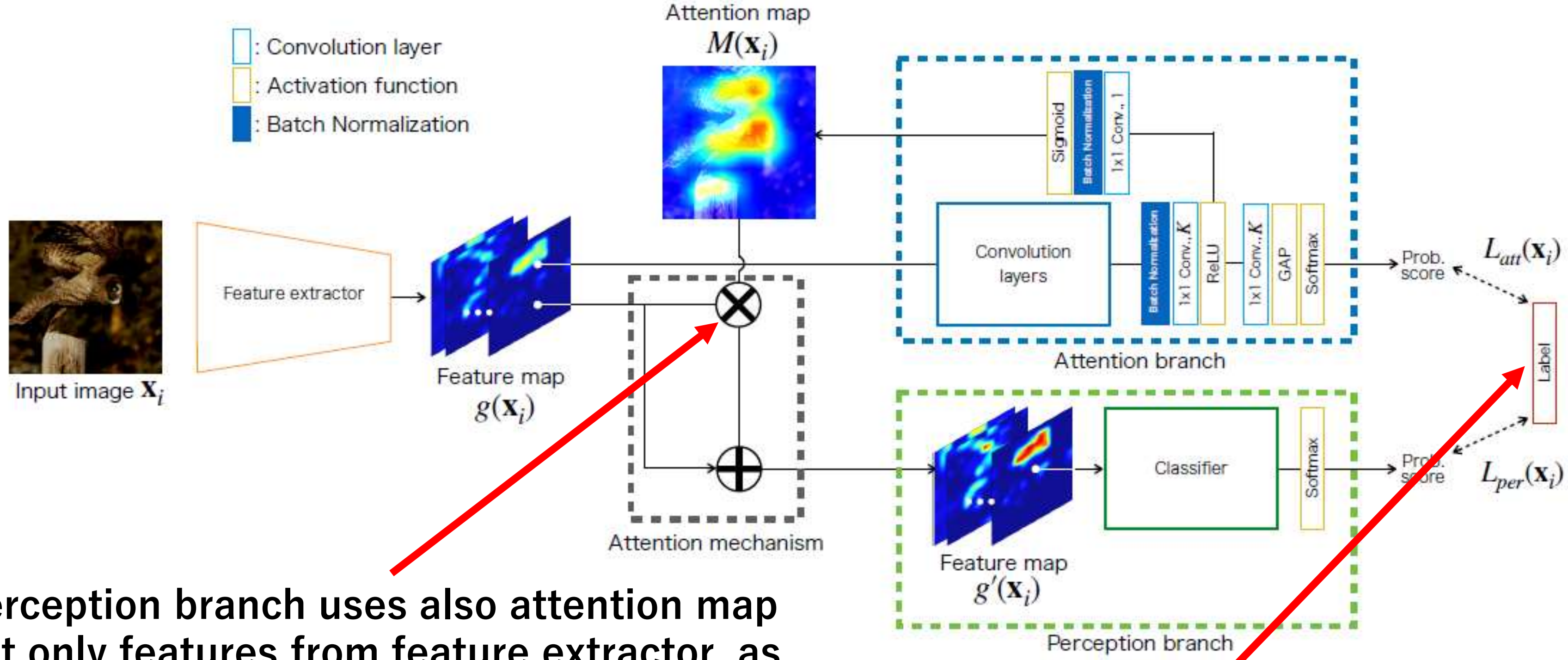
(a) Class Activation Mapping



(b) Attention Branch Network

Can add Fc layers!!

Method 2: ABN for image classification



Perception branch uses also attention map not only features from feature extractor, as its input.

$$g'_c(\mathbf{x}_i) = M(\mathbf{x}_i) \cdot g_c(\mathbf{x}_i)$$

Loss function has attention/perception terms.
Both consists of cross entropy for classification task.

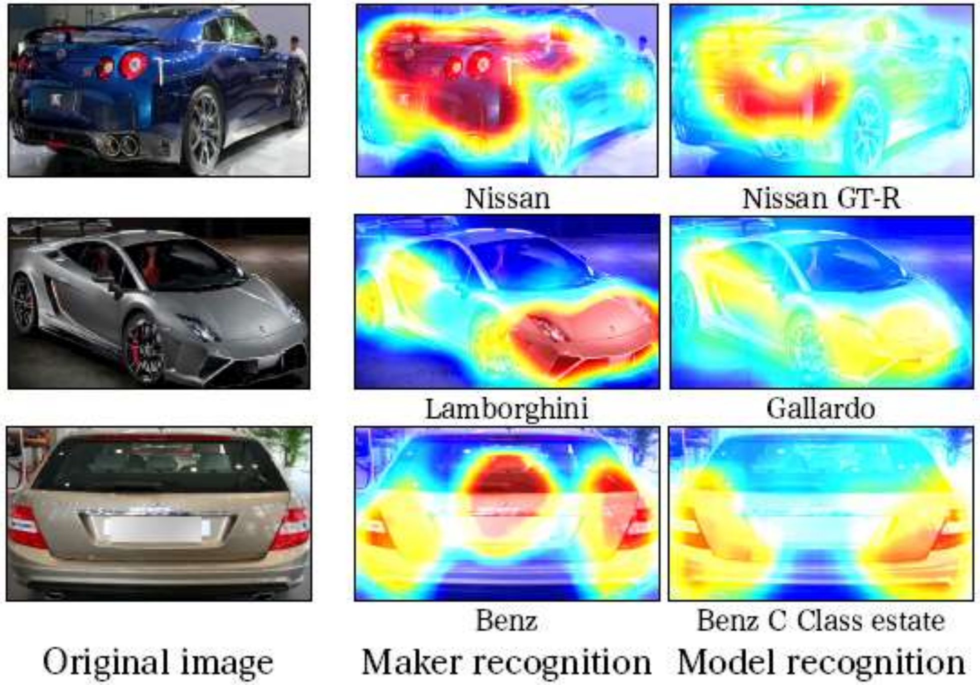
$$L(\mathbf{x}_i) = L_{att}(\mathbf{x}_i) + L_{per}(\mathbf{x}_i)$$

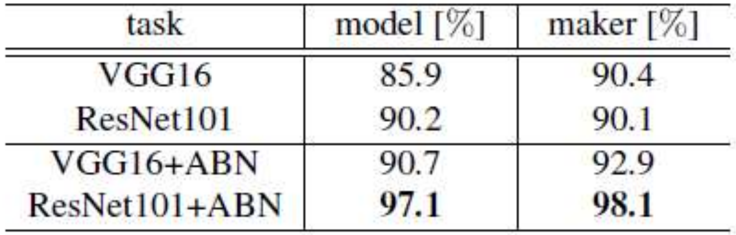
☺

- Improved classification accuracy because it can use Fc layers.
- Actually, using attention map in the input of perception/loss function improves accuracy.

Method 2: ABN for image classification

Results of ABN





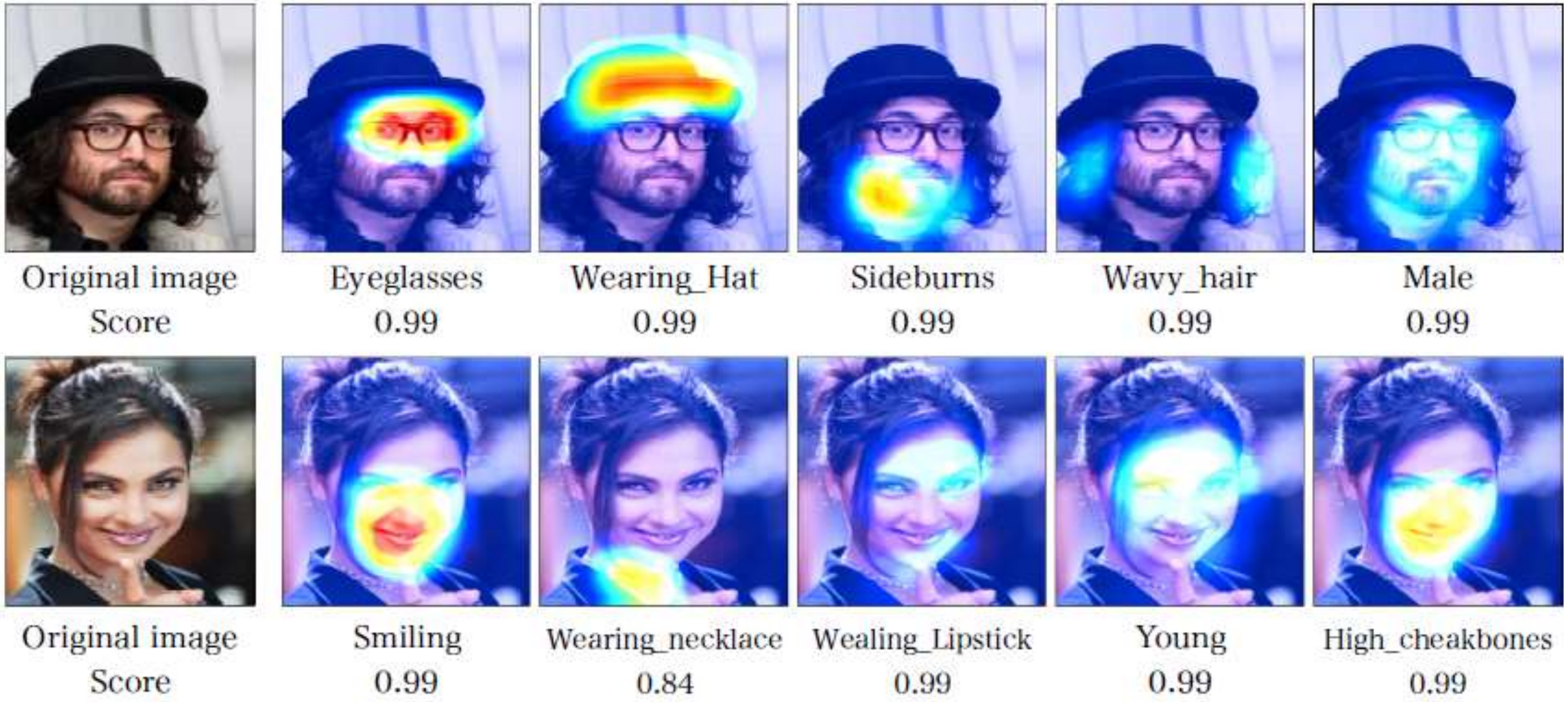
task	model [%]	maker [%]
VGG16	85.9	90.4
ResNet101	90.2	90.1
VGG16+ABN	90.7	92.9
ResNet101+ABN	**97.1**	**98.1**

Attention map improves accuracy!

As I do not have domain knowledge,
I can not judge whether the model is focusing on correct features…
Are the highlighted parts characteristic for each maker/model?

It is interesting that, the attention is paid to different parts depending on the task.

Method 2: ABN for image classification



The model seems to be focusing on correct features.

Method 2: ABN for image classification

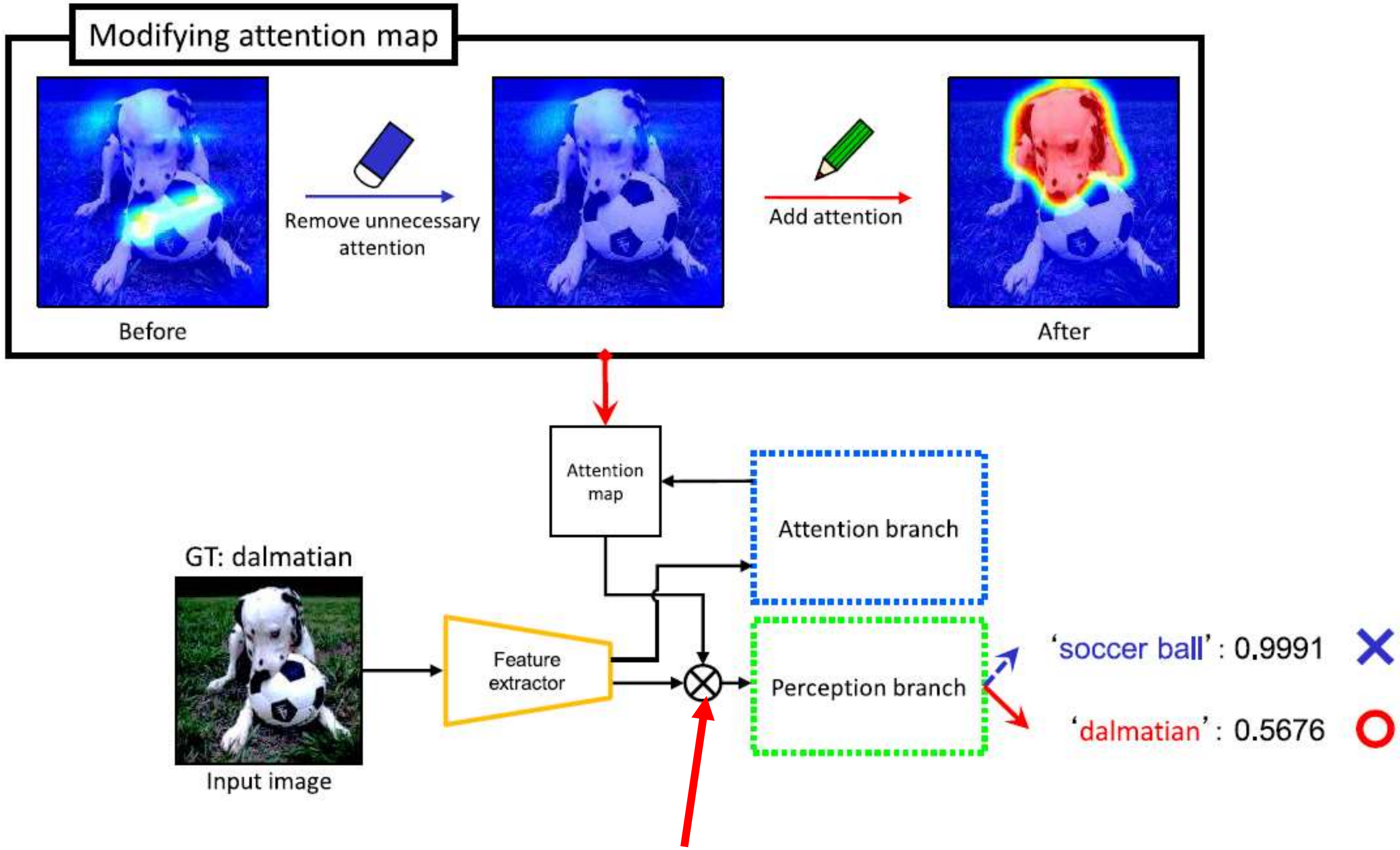
ABN and human-in-the-loop

M. Mitsuhara et. al., arXiv: 1905.03540

Basic idea: By modifying attention map in ABN using human knowledge, try to improve the accuracy of image classifier.

Method 2: ABN for image classification

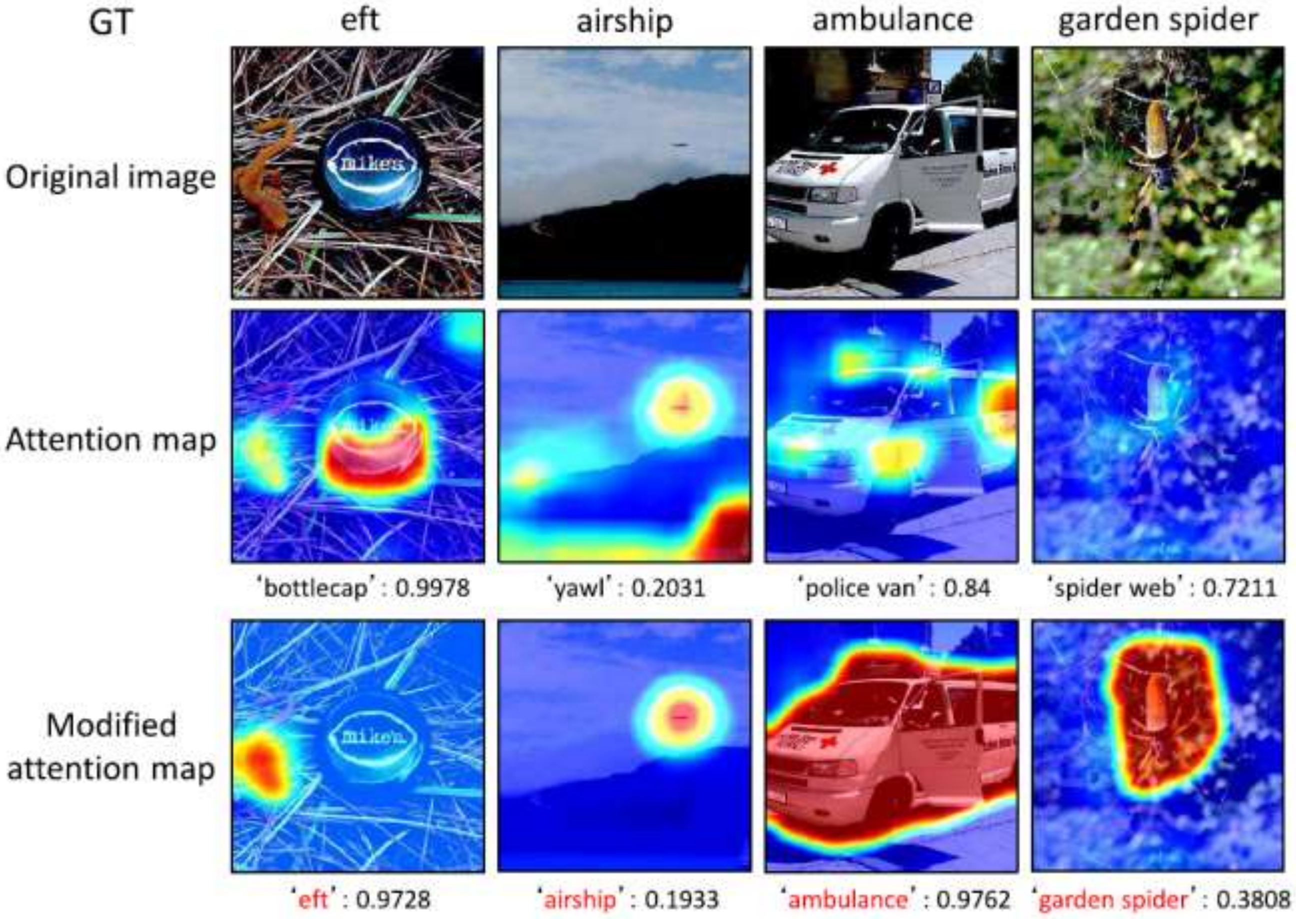
Modifying attention map manually helps classification



Perception branch uses also attention map not only features from feature extractor, as its input.

Method 2: ABN for image classification

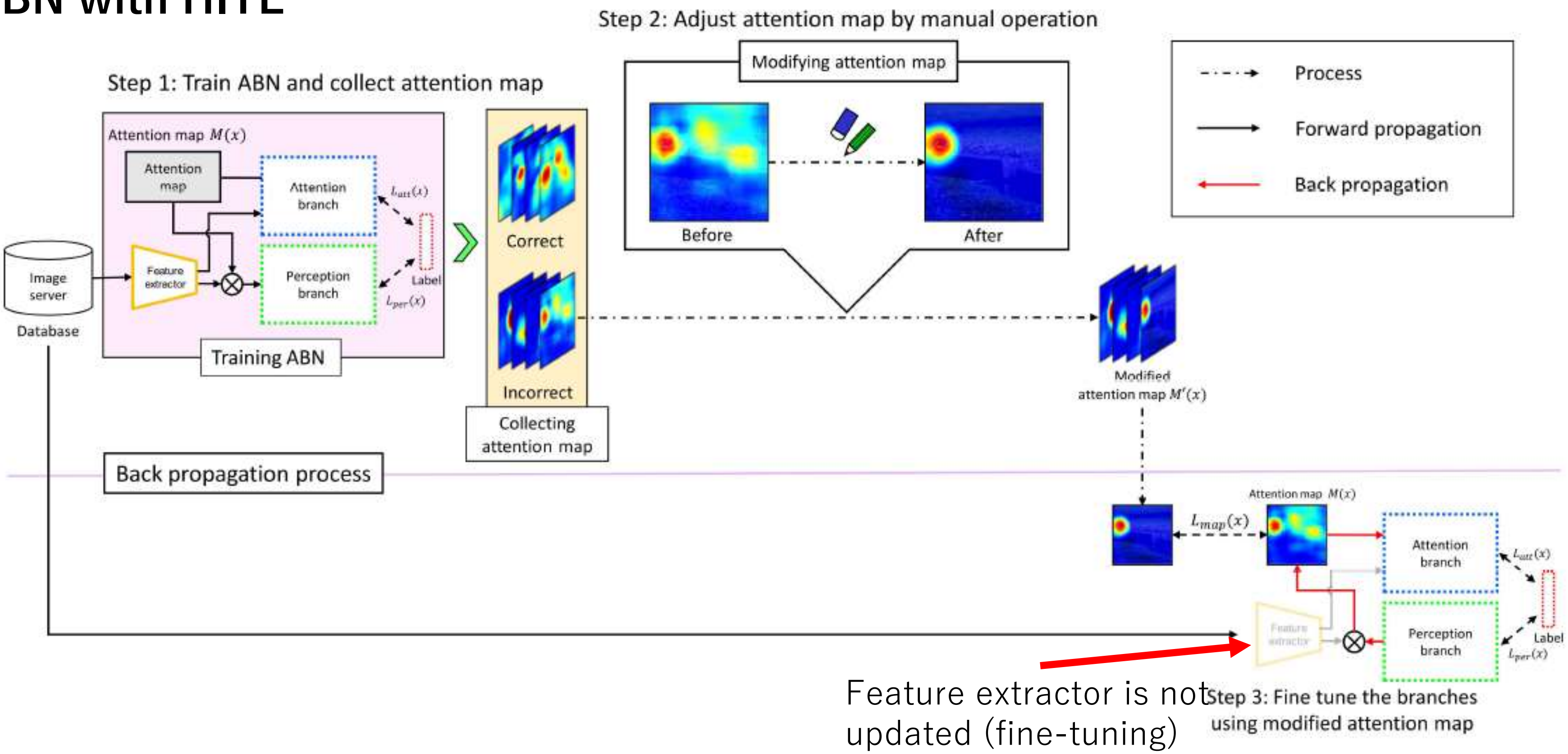
Results



Modifying the attention map improved the classification accuracy

Method 2: ABN for image classification

ABN with HITL



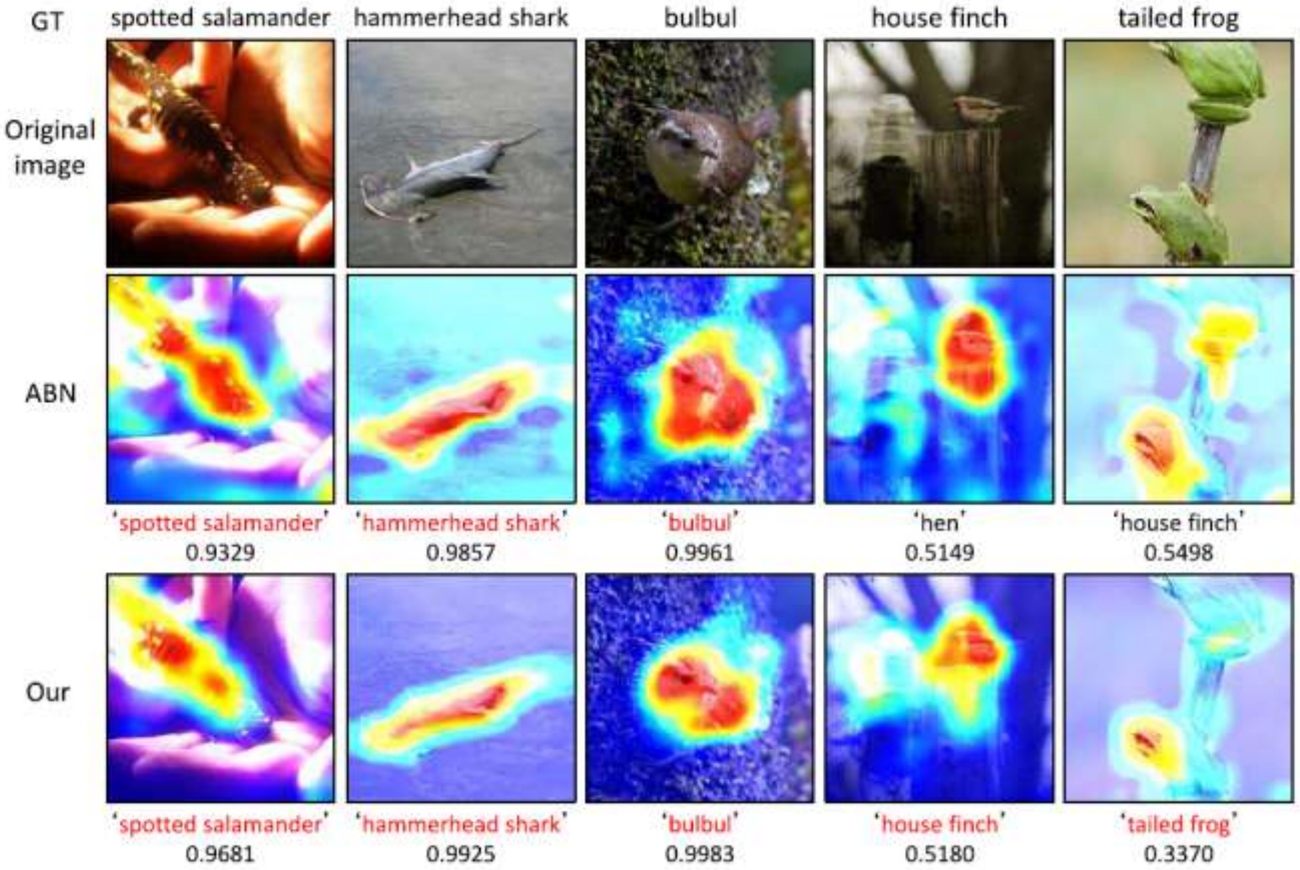
Feature extractor is not updated (fine-tuning)

$$L_{all}(\mathbf{x}) = L_{att}(\mathbf{x}) + L_{per}(\mathbf{x}) + L_{map}(\mathbf{x})$$

$$L_{map}(\mathbf{x}) = \gamma \| M'(\mathbf{x}) - M(\mathbf{x}) \|_2$$

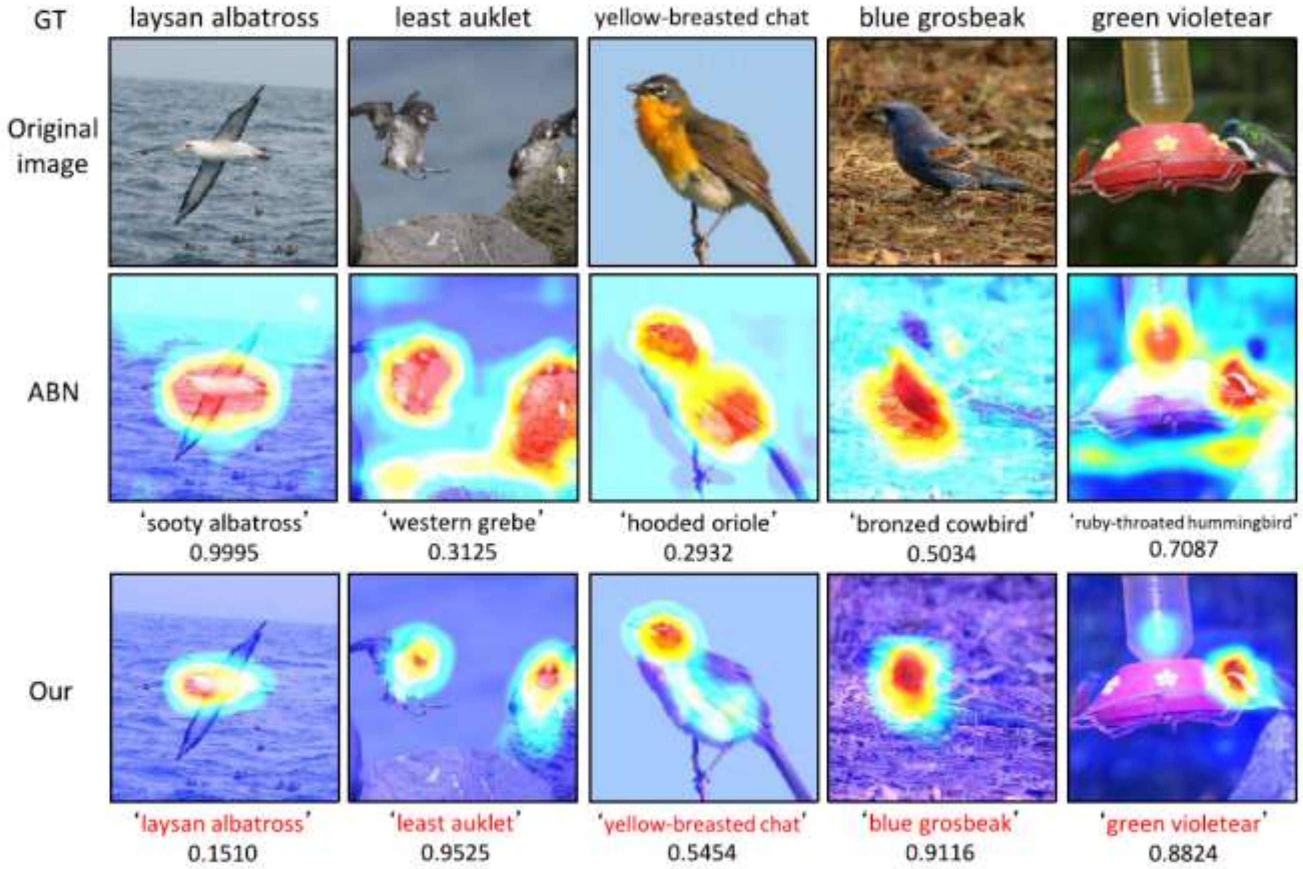
Method 2: ABN for image classification

Results



HITL improves the attention map (not very apparent), and also classification accuracy.

Method 2: ABN for image classification



HITL improves the attention map (focusing on relevant parts, not the whole body), and also classification accuracy.

Method 2: ABN for image classification

HITL is possible because the explanation (attention map) is given as a part of their model in ABN, and it is reused as input of perception branch. Unlike LIME, ABN is not model-agnostic.

So maybe, being model-agnostic is not always useful: Sometimes it is better to be able to touch the model.

Summary

I introduced a few recent methods on XAI.

1. **LIME**

   for

   - Classification of structured data
   - Regression of structured data
   - Classification of image

2. **ABN**

   for

   - Classification of image

I also introduced **application of human-in-the-loop to ABN**

Arithmer 株式会社
〒106-6040
東京都港区六本木一丁目６番１号 泉ガーデンタワー 38/40F(受付)
03-5579-6683
https://arithmer.co.jp/

Arithmer