Explainable Deep learning

Anirudh Singh, Ankur Debnath, Deep Patel, Lalit Manam

E0-270 Machine Learning

April 27, 2019

Motivation

- Need for understanding models instead of treating them as black box
- Trusting a prediction i.e. whether a user trusts an individual prediction sufficiently to take some action based on it
- Trusting a model i.e. whether the user trusts a model to behave in reasonable ways if deployed

Motivation

- Need for understanding models instead of treating them as black box
- Trusting a prediction i.e. whether a user trusts an individual prediction sufficiently to take some action based on it
- Trusting a model i.e. whether the user trusts a model to behave in reasonable ways if deployed

Motivation

- Need for understanding models instead of treating them as black box
- Trusting a prediction i.e. whether a user trusts an individual prediction sufficiently to take some action based on it
- Trusting a model i.e. whether the user trusts a model to behave in reasonable ways if deployed

Literature Survey

Few relevant papers:

- Model compression
- Unifying distillation and privileged information
- Learning global additive explanations for neural nets using model distillation
- Born-again neural networks
- Hierarchical interpretations for neural network predictions
- Distilling a neural network into a soft decision tree

Model compression

Proposed by Bucilua[1]

- Train fast and compact neural nets to mimic function learned by ensemble selection/larger model
- More data is required in training phase for simpler model to achieve the accuracy rates nearby the complex model
- Pseudo data: Use of the large complex model to label data. Direct use of unlabelled data or creation of synthetic data

[1] Cristian Bucilu, Rich Caruana, and Alexandru Niculescu-Mizil. "Model compression". In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pp. 535–541.

Model compression

Proposed by Bucilua[1]

- Train fast and compact neural nets to mimic function learned by ensemble selection/larger model
- More data is required in training phase for simpler model to achieve the accuracy rates nearby the complex model
- Pseudo data: Use of the large complex model to label data. Direct use of unlabelled data or creation of synthetic data

[1] Bucilu, Caruana, and Niculescu-Mizil, "Model compression".

Model compression

Proposed by Bucilua[1]

- Train fast and compact neural nets to mimic function learned by ensemble selection/larger model
- More data is required in training phase for simpler model to achieve the accuracy rates nearby the complex model
- Pseudo data: Use of the large complex model to label data. Direct use of unlabelled data or creation of synthetic data

[1] Bucilu, Caruana, and Niculescu-Mizil, "Model compression".

Model compression

Proposed by Bucilua[1]

- Train fast and compact neural nets to mimic function learned by ensemble selection/larger model
- More data is required in training phase for simpler model to achieve the accuracy rates nearby the complex model
- Pseudo data: Use of the large complex model to label data. Direct use of unlabelled data or creation of synthetic data

[1] Bucilu, Caruana, and Niculescu-Mizil, "Model compression".

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
    - Creation of soft labels from the big model (modified soft-max)
    - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
    - Limited to SVMs initially
    - Teacher function estimates the slack values used in objective function
    - Teacher function finds best set of prototype points using privileged information
    - Student function learns using these prototype points
- Generalized distillation:
    - Learn teacher-student model with privileged information
    - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Unifying distillation and privileged information

Proposed by Lopez-Paz[2]

- Uses generalized distillation combining distillation and usage of privileged information
- Distillation:
  - Creation of soft labels from the big model (modified soft-max)
  - Objective function balances imitating both the soft and hard labels appropriately
- Using privileged information:
  - Limited to SVMs initially
  - Teacher function estimates the slack values used in objective function
  - Teacher function finds best set of prototype points using privileged information
  - Student function learns using these prototype points
- Generalized distillation:
  - Learn teacher-student model with privileged information
  - Distillation process to train student using hard and soft labels

Learning global additive explanations for neural nets using model distillation

Proposed by Tan[3]

- Distillation process is carried on by matching the logits of the original/complex model
- Tan considers learning the linear combination of feature maps
- Additive terms represent feature shapes (contribution of a feature across the entire domain) which are better global descriptor than feature attribution (features contribution to either the prediction of one sample)
- Soft labels obtained from teacher model are used to train the student model i.e. feature maps

[3] Sarah Tan et al. "Learning Global Additive Explanations for Neural Nets Using Model Distillation". In: (2018).

Learning global additive explanations for neural nets using model distillation

Proposed by Tan[3]

- Distillation process is carried on by matching the logits of the original/complex model
- Tan considers learning the linear combination of feature maps
- Additive terms represent feature shapes (contribution of a feature across the entire domain) which are better global descriptor than feature attribution (features contribution to either the prediction of one sample)
- Soft labels obtained from teacher model are used to train the student model i.e. feature maps

[3] Tan et al., "Learning Global Additive Explanations for Neural Nets Using Model Distillation".

Learning global additive explanations for neural nets using model distillation

Proposed by Tan[3]

- Distillation process is carried on by matching the logits of the original/complex model
- Tan considers learning the linear combination of feature maps
- Additive terms represent feature shapes (contribution of a feature across the entire domain) which are better global descriptor than feature attribution (features contribution to either the prediction of one sample)
- Soft labels obtained from teacher model are used to train the student model i.e. feature maps

[3] Tan et al., "Learning Global Additive Explanations for Neural Nets Using Model Distillation".

Learning global additive explanations for neural nets using model distillation

Proposed by Tan[3]

- Distillation process is carried on by matching the logits of the original/complex model
- Tan considers learning the linear combination of feature maps
- Additive terms represent feature shapes (contribution of a feature across the entire domain) which are better global descriptor than feature attribution (features contribution to either the prediction of one sample)
- Soft labels obtained from teacher model are used to train the student model i.e. feature maps

[3] Tan et al., "Learning Global Additive Explanations for Neural Nets Using Model Distillation".

Born again neural networks

Proposed by Furlanello[4]

- Variant of knowledge distillation where the purpose is to exact better performance from the distilled models
- Models are trained in sequence, each one from the last and finally ensemble averaging is performed
- Trained models of similar capacity outperform their teachers
- Error gradient can be split into ground truth and dark knowledge components, both are back-propagated
- Superior performance is often attributed to the dark knowledge part of model distillation

[4] Tommaso Furlanello et al. "Born-Again Neural Networks". In: *International Conference on Machine Learning*. 2018, pp. 1602–1611.

Born again neural networks

Proposed by Furlanello[4]

- Variant of knowledge distillation where the purpose is to exact better performance from the distilled models
- Models are trained in sequence, each one from the last and finally ensemble averaging is performed
- Trained models of similar capacity outperform their teachers
- Error gradient can be split into ground truth and dark knowledge components, both are back-propagated
- Superior performance is often attributed to the dark knowledge part of model distillation

[4] Furlanello et al., "Born-Again Neural Networks".

Born again neural networks

Proposed by Furlanello[4]

- Variant of knowledge distillation where the purpose is to exact better performance from the distilled models
- Models are trained in sequence, each one from the last and finally ensemble averaging is performed
- Trained models of similar capacity outperform their teachers
- Error gradient can be split into ground truth and dark knowledge components, both are back-propagated
- Superior performance is often attributed to the dark knowledge part of model distillation

[4] Furlanello et al., "Born-Again Neural Networks".

Born again neural networks

Proposed by Furlanello[4]

- Variant of knowledge distillation where the purpose is to exact better performance from the distilled models
- Models are trained in sequence, each one from the last and finally ensemble averaging is performed
- Trained models of similar capacity outperform their teachers
- Error gradient can be split into ground truth and dark knowledge components, both are back-propagated
- Superior performance is often attributed to the dark knowledge part of model distillation

[4] Furlanello et al., "Born-Again Neural Networks".

Born again neural networks

Proposed by Furlanello[4]

- Variant of knowledge distillation where the purpose is to exact better performance from the distilled models
- Models are trained in sequence, each one from the last and finally ensemble averaging is performed
- Trained models of similar capacity outperform their teachers
- Error gradient can be split into ground truth and dark knowledge components, both are back-propagated
- Superior performance is often attributed to the dark knowledge part of model distillation

[4] Furlanello et al., "Born-Again Neural Networks".

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Chandan Singh, W James Murdoch, and Bin Yu. "Hierarchical interpretations for neural network predictions". In: *arXiv preprint arXiv:1806.05337* (2018).

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Singh, Murdoch, and Yu, "Hierarchical interpretations for neural network predictions".

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Singh, Murdoch, and Yu, "Hierarchical interpretations for neural network predictions".

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Singh, Murdoch, and Yu, "Hierarchical interpretations for neural network predictions".

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Singh, Murdoch, and Yu, "Hierarchical interpretations for neural network predictions".

Hierarchical interpretations for neural network predictions

Proposed by Singh[5]

- Agglomerative Contextual Decompositions(ACD)
- Hierarchical clustering of groups of input features
- Aims to capture the interaction between features that a DNN has learned while striking a balance between simplicity and information contained
- Yields a subset of groups of features that are both indicative of the interaction between features and compact enough not to be overwhelming
- Idea of contextual decompositions generalized from LSTMs to generic DNNs
- Hierarchical saliency : A group level importance measure is used as joining metric for contextual agglomerative clustering

[5] Singh, Murdoch, and Yu, "Hierarchical interpretations for neural network predictions".

Distilling a Neural Network Into a Soft Decision Tree

Proposed by Frosst[6]

- Inspired by the hierarchical mixture of experts model (Jordan et al. (1994))
- Soft decision tree (DT) uses the learned filters to make hierarchical decisions for an instance
- DT offers explanability of classification decision unlike in case of NNs.
- Explanability at the expense of performance degradation

[6] Nicholas Frosst and Geoffrey Hinton. "Distilling a neural network into a soft decision tree". In: *arXiv preprint arXiv:1711.09784* (2017).

Distilling a Neural Network Into a Soft Decision Tree

Proposed by Frosst[6]

- Inspired by the hierarchical mixture of experts model (Jordan et al. (1994))
- Soft decision tree (DT) uses the learned filters to make hierarchical decisions for an instance
- DT offers explanability of classification decision unlike in case of NNs.
- Explanability at the expense of performance degradation

[6] Frosst and Hinton, "Distilling a neural network into a soft decision tree".

Distilling a Neural Network Into a Soft Decision Tree

Proposed by Frosst[6]

- Inspired by the hierarchical mixture of experts model (Jordan et al. (1994))
- Soft decision tree (DT) uses the learned filters to make hierarchical decisions for an instance
- DT offers explanability of classification decision unlike in case of NNs.
- Explanability at the expense of performance degradation

[6] Frosst and Hinton, "Distilling a neural network into a soft decision tree".

Distilling a Neural Network Into a Soft Decision Tree

Proposed by Frosst[6]

- Inspired by the hierarchical mixture of experts model (Jordan et al. (1994))
- Soft decision tree (DT) uses the learned filters to make hierarchical decisions for an instance
- DT offers explanability of classification decision unlike in case of NNs.
- Explanability at the expense of performance degradation

[6] Frosst and Hinton, "Distilling a neural network into a soft decision tree".

Our work

Our work considers the paper proposed by Ribeiro et. al.[7].

[7] Marco Tulio Ribeiro, 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*. ACM. 2016, pp. 1135–1144.

Local Interpretable Model-Agnostic Explanations

Explanation produced by LIME:

$$\xi(x) = \text{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \quad (1)$$

- Explanation defined as a model $g \in G$, G is the set of interpretable models
- $\Omega(g)$ - measure of complexity (as opposed to interpretability) of the explanation g
- $f : \Re^d \rightarrow \Re$ - Model being explanied
- $\pi_x(z)$ - proximity measure between an instance z to x
- $\mathcal{L}(f, g, \pi_x)$ - Loss function

Local Interpretable Model-Agnostic Explanations

Explanation produced by LIME:

$$\xi(x) = \text{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

- Explanation defined as a model $g \in G$, G is the set of interpretable models
- $\Omega(g)$ - measure of complexity (as opposed to interpretability) of the explanation g
- $f : \Re^d \rightarrow \Re$ - Model being explanied
- $\pi_x(z)$ - proximity measure between an instance z to x
- $\mathcal{L}(f, g, \pi_x)$ - Loss function

Local Interpretable Model-Agnostic Explanations

Explanation produced by LIME:

$$\xi(x) = \text{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

- Explanation defined as a model $g \in G$, G is the set of interpretable models
- $\Omega(g)$ - measure of complexity (as opposed to interpretability) of the explanation g
- $f : \Re^d \rightarrow \Re$ - Model being explanied
- $\pi_x(z)$ - proximity measure between an instance z to x
- $\mathcal{L}(f, g, \pi_x)$ - Loss function

Local Interpretable Model-Agnostic Explanations

Explanation produced by LIME:

$$\xi(x) = \text{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

- Explanation defined as a model $g \in G$, G is the set of interpretable models
- $\Omega(g)$ - measure of complexity (as opposed to interpretability) of the explanation g
- $f : \Re^d \rightarrow \Re$ - Model being explanied
- $\pi_x(z)$ - proximity measure between an instance z to x
- $\mathcal{L}(f, g, \pi_x)$ - Loss function

Local Interpretable Model-Agnostic Explanations

Explanation produced by LIME:

$$\xi(x) = \text{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

- Explanation defined as a model $g \in G$, G is the set of interpretable models
- $\Omega(g)$ - measure of complexity (as opposed to interpretability) of the explanation g
- $f : \Re^d \rightarrow \Re$ - Model being explanied
- $\pi_x(z)$ - proximity measure between an instance z to x
- $\mathcal{L}(f, g, \pi_x)$ - Loss function

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - General case

- Approximate $\mathcal{L}(f, g, \pi_x)$ by drawing samples weighted by π_x
- Given x' (explaiable domain) by drawing nonzero elements of x' uniformly at random, call it perturbed sample z'
- Given z' recover original representation (data domain)
- Obtain $f(z)$, which is used as a label for the explanation
- Collection of z' leads to a dataset $\mathcal{Z}$
- Use $\mathcal{Z}$ to solve (1)

Local Interpretable Model-Agnostic Explanations

LIME - Sparse linear models

- G is a class of linear models such that $g(z') = w_g \cdot z'$
- Locally weighted square loss as $\mathcal{L}$, where $\pi_x(z) = exp(-D(x,z)^2/\sigma^2)$
  - D is some distance function

Loss function:

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

Local Interpretable Model-Agnostic Explanations

LIME - Sparse linear models

- G is a class of linear models such that $g(z') = w_g \cdot z'$
- Locally weighted square loss as $\mathcal{L}$, where $\pi_x(z) = exp(-D(x,z)^2/\sigma^2)$
  - D is some distance function

Loss function:

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

Local Interpretable Model-Agnostic Explanations

LIME - Sparse linear models

- G is a class of linear models such that $g(z') = w_g \cdot z'$
- Locally weighted square loss as $\mathcal{L}$, where $\pi_x(z) = exp(-D(x,z)^2/\sigma^2)$
    - D is some distance function

Loss function:

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

Local Interpretable Model-Agnostic Explanations

LIME - Sparse linear models

- G is a class of linear models such that $g(z') = w_g \cdot z'$
- Locally weighted square loss as $\mathcal{L}$, where $\pi_x(z) = exp(-D(x,z)^2/\sigma^2)$
  - D is some distance function

Loss function:

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

Local Interpretable Model-Agnostic Explanations

LIME Algorithm for Sparse Linear Explanations

Require: Classifier f, Number of samples N
Require: Instance x, and its interpretable version x'
Require: Similarity Kernel π_x, Length of explanation K
$\mathcal{Z} \leftarrow \{\}$ (perturbed samples)
for $i = 1$ **to** N **do**
  $z'_i \leftarrow$ sample around(x')
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z'_i, f(z_i), \pi_x(z_i) \rangle$
end for
$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$
with z'_i as features, $f(z)$ as target .
return w

Local Interpretable Model-Agnostic Explanations

LIME Algorithm for Sparse Linear Explanations

Require: Classifier f, Number of samples N
Require: Instance x, and its interpretable version x'
Require: Similarity Kernel π_x, Length of explanation K
$\mathcal{Z} \leftarrow \{\}$ (perturbed samples)
for $i = 1$ **to** N **do**
  $z_i' \leftarrow$ sample around(x')
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z_i', f(z_i), \pi_x(z_i) \rangle$
end for
$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$
with z_i' as features, $f(z)$ as target .
return w

Local Interpretable Model-Agnostic Explanations

LIME Algorithm for Sparse Linear Explanations

Require: Classifier f, Number of samples N
Require: Instance x, and its interpretable version x'
Require: Similarity Kernel π_x, Length of explanation K
$\mathcal{Z} \leftarrow \{\}$ (perturbed samples)
for $i = 1$ **to** N **do**
  $z'_i \leftarrow$ sample around(x')
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z'_i, f(z_i), \pi_x(z_i) \rangle$
end for
$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$
with z'_i as features, $f(z)$ as target .
return w

Need for extension of LIME

- LIME generates local explanations
- Infeasible to check local explanations for each instance of dataset
- Local explanations need not be indicators of global behaviour
- Need global explanations to explain the behaviour of a model (classifier)

Need for extension of LIME

- LIME generates local explanations
- Infeasible to check local explanations for each instance of dataset
- Local explanations need not be indicators of global behaviour
- Need global explanations to explain the behaviour of a model (classifier)

Need for extension of LIME

- LIME generates local explanations
- Infeasible to check local explanations for each instance of dataset
- Local explanations need not be indicators of global behaviour
- Need global explanations to explain the behaviour of a model (classifier)

Need for extension of LIME

- LIME generates local explanations
- Infeasible to check local explanations for each instance of dataset
- Local explanations need not be indicators of global behaviour
- Need global explanations to explain the behaviour of a model (classifier)

Sub-Modular Pick

- Construct an explanation matrix, $W = \mathbb{R}^{n \times d}$, for a set of instances X
  - $n = \#instances$
  - $d = \#features$
- For the sparse linear model (g_i) as described earlier, choose $W_{ij} = |w_{g_{ij}}|$
- Compute the global importance (I_j) for each j in W
- I reflects importance of the features appearing prominently in the local explanations of the instances
- Authors advocate use of $I_j = \sqrt{\sum_i W_{ij}}$

Sub-Modular Pick

- Construct an explanation matrix, $W = \mathbb{R}^{n \times d}$, for a set of instances X
  - $n = \#instances$
  - $d = \#features$
- For the sparse linear model (g_i) as described earlier, choose $W_{ij} = |w_{g_{ij}}|$
- Compute the global importance (I_j) for each j in W
- I reflects importance of the features appearing prominently in the local explanations of the instances
- Authors advocate use of $I_j = \sqrt{\sum_i W_{ij}}$

Sub-Modular Pick

- Construct an explanation matrix, $W = \mathbb{R}^{n \times d}$, for a set of instances X
  - $n = \#instances$
  - $d = \#features$
- For the sparse linear model (g_i) as described earlier, choose $W_{ij} = |w_{g_{ij}}|$
- Compute the global importance (I_j) for each j in W
- I reflects importance of the features appearing prominently in the local explanations of the instances
- Authors advocate use of $I_j = \sqrt{\sum_i W_{ij}}$

Sub-Modular Pick

- Construct an explanation matrix, $W = \mathbb{R}^{n \times d}$, for a set of instances X
  - $n = \#instances$
  - $d = \#features$
- For the sparse linear model (g_i) as described earlier, choose $W_{ij} = |w_{g_{ij}}|$
- Compute the global importance (I_j) for each j in W
- I reflects importance of the features appearing prominently in the local explanations of the instances
- Authors advocate use of $I_j = \sqrt{\sum_i W_{ij}}$

Sub-Modular Pick

- Construct an explanation matrix, $W = \mathbb{R}^{n \times d}$, for a set of instances X
  - $n = \#instances$
  - $d = \#features$
- For the sparse linear model (g_i) as described earlier, choose $W_{ij} = |w_{g_{ij}}|$
- Compute the global importance (I_j) for each j in W
- I reflects importance of the features appearing prominently in the local explanations of the instances
- Authors advocate use of $I_j = \sqrt{\sum_i W_{ij}}$

Sub-Modular Pick

- c computes the total importance of the features that appear in at least one instance in the set V.

$$c(V, W, I) = \sum_{j=1}^{d'} 1_{[\exists i \in V : W_{ij} > 0]} I_j \tag{3}$$

- The pick problem consists finding V, such that $|V| \leq B$ that achieves highest coverage (c).

$$Pick(W, I) = \operatorname{argmax}_{V, |V| \leq B} c(V, W, I) \tag{4}$$

The problem in (4) is maximizing a weighted coverage function and is also NP-hard. Thus, a greedy strategy as outlined in algorithm 18 is employed.

Sub-Modular Pick

- c computes the total importance of the features that appear in at least one instance in the set V.

$$c(V, W, I) = \sum_{j=1}^{d'} 1_{[\exists i \in V : W_{ij} > 0]} I_j \tag{3}$$

- The pick problem consists finding V, such that $|V| \leq B$ that achieves highest coverage (c).

$$Pick(W, I) = \text{argmax}_{V, |V| \leq B} \, c(V, W, I) \tag{4}$$

The problem in (4) is maximizing a weighted coverage function and is also NP-hard. Thus, a greedy strategy as outlined in algorithm 18 is employed.

Algorithm for Sub-Modular Pick

Require: Instances X, Budget B
for $x_i \in X$ **do**
  $W_i \leftarrow \text{explain}(x_i, x_i')$ (Use algorithm 14)
end for
for $j \in \{0...d'\}$ **do**
  $I_j \leftarrow \sqrt{\sum_i^n |W_{ij}|}$ (Compute feature importance)
end for
$V \leftarrow \{\}$
(Next step is greedy optimization of (4))
while $|V| < B$ **do**
  $V \leftarrow V \cup \text{argmax}_i\, c(V \cup \{i\}, W, I)$
end while
return Return V

Algorithm for Sub-Modular Pick

Require: Instances X, Budget B
for $x_i \in X$ **do**
  $W_i \leftarrow \text{explain}(x_i, x_i')$ (Use algorithm 14)
end for
for $j \in \{0 \ldots d'\}$ **do**
  $I_j \leftarrow \sqrt{\sum_i^n |W_{ij}|}$ (Compute feature importance)
end for
$V \leftarrow \{\}$
(Next step is greedy optimization of (4))
while $|V| < B$ **do**
  $V \leftarrow V \cup \text{argmax}_i \, c(V \cup \{i\}, W, I)$
end while
return Return V

Algorithm for Sub-Modular Pick

Require: Instances X, Budget B
for $x_i \in X$ **do**
  $W_i \leftarrow \text{explain}(x_i, x_i')$ (Use algorithm 14)
end for
for $j \in \{0...d'\}$ **do**
  $I_j \leftarrow \sqrt{\sum_i^n |W_{ij}|}$ (Compute feature importance)
end for
$V \leftarrow \{\}$
(Next step is greedy optimization of (4))
while $|V| < B$ **do**
  $V \leftarrow V \cup \text{argmax}_i\, c(V \cup \{i\}, W, I)$
end while
return **Return** V

Algorithm for Sub-Modular Pick

Require: Instances X, Budget B
for $x_i \in X$ **do**
  $W_i \leftarrow \text{explain}(x_i, x_i')$ (Use algorithm 14)
end for
for $j \in \{0 \ldots d'\}$ **do**
  $I_j \leftarrow \sqrt{\sum_i^n |W_{ij}|}$ (Compute feature importance)
end for
$V \leftarrow \{\}$
(Next step is greedy optimization of (4))
while $|V| < B$ **do**
  $V \leftarrow V \cup \text{argmax}_i \, c(V \cup \{i\}, W, I)$
end while
return **Return** V

Experimental Setup

- The pipeline as shown in Figure 1 describes the procedure to replicate the results of Ribeiro[8]
- Train two different classifiers (neural networks) for a classification task on a certain dataset after which, the LIME package (provided by the authors) is used to generate local explanations for a given instance from the dataset.
- LIME are local, hence sub-modular picking is needed to generate global explanations
- These explanations are given to human subjects for a trustworthiness test.

[8] Ribeiro, Singh, and Guestrin, "Why should i trust you?: Explaining the predictions of any classifier".

Experimental Setup

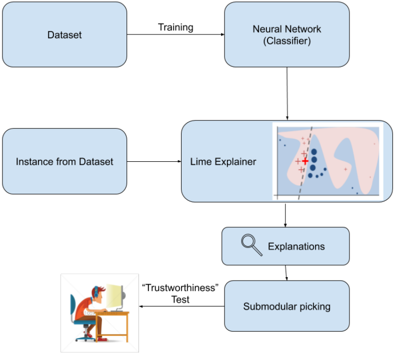
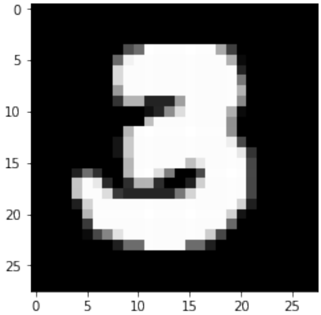


Figure: Pipeline for generating explanation from a classifier

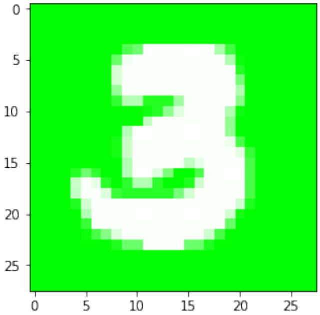
Setup

- For the image classification explanability task, we have trained a CNN on MNIST and CIFAR-10 image dataset.
- Since the fine tuning of networks does not play a role in explanability, the details of the architecture used here are omitted.
- LIME generated for some of the instances are shown in the next few slides.

Results: LIME Explanations for MNIST



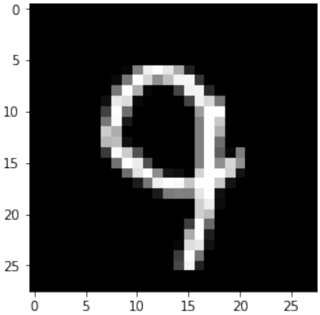
(a) Original instance from MNIST dataset



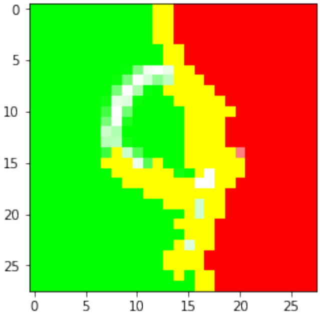
(b) Local explanation for correct classification

Figure: LIME explanations for MNIST dataset instance

Results: LIME Explanations for MNIST



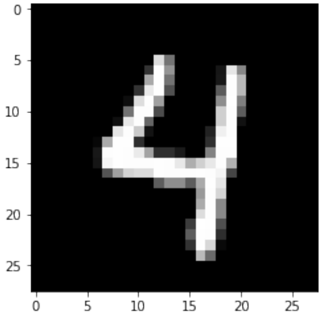
(a) Original instance from MNIST dataset



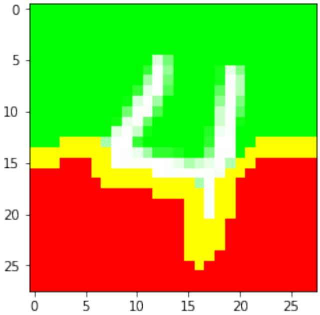
(b) Local explanation for correct classification

Figure: LIME explanations for MNIST dataset instance

Results: LIME Explanations for MNIST



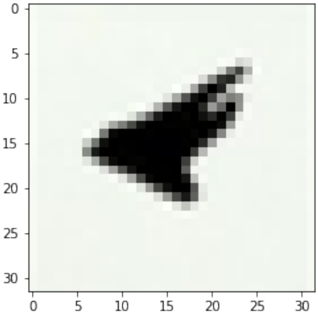
(a) Original instance from MNIST dataset



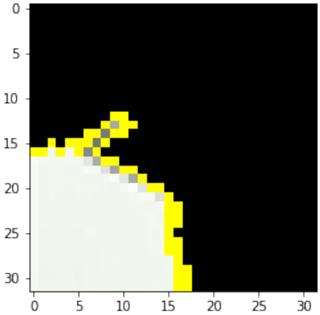
(b) Local explanation for correct classification

Figure: LIME explanations for MNIST dataset instance

Results: LIME Explanations for CIFAR-10



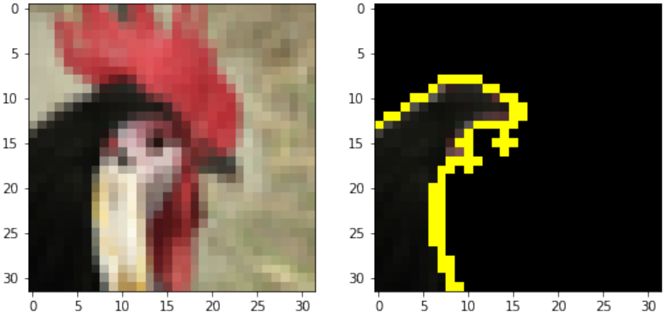
(a) Original instance from CIFAR-10 dataset



(b) Local explanation for misclassification as BIRD

Figure: LIME explanations for CIFAR-10 dataset instance

Results: LIME Explanations for CIFAR-10

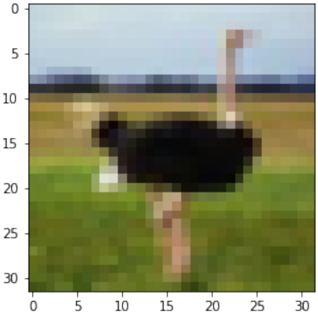


(a) Original instance from CIFAR-10 dataset

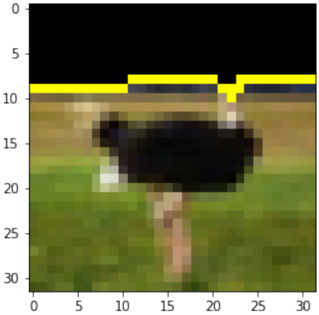
(b) Local explanation for correct classification as BIRD

Figure: LIME explanations for CIFAR-10 dataset instance

Results: LIME Explanations for CIFAR-10



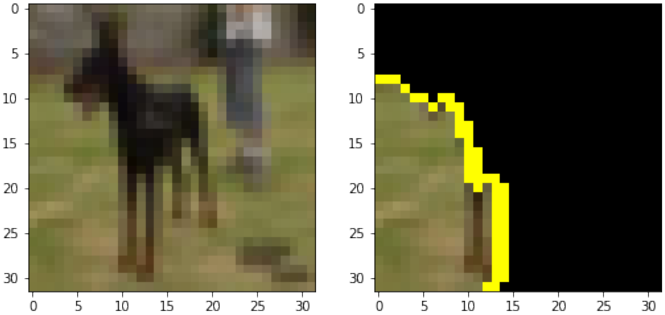
(a) Original instance from CIFAR-10 dataset



(b) Local explanation for correct classification as BIRD

Figure: LIME explanations for CIFAR-10 dataset instance

Results: LIME Explanations for CIFAR-10

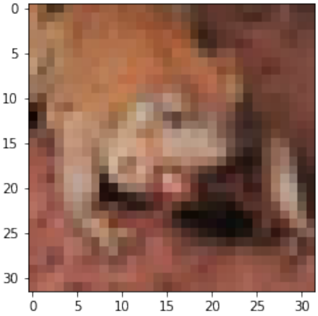


(a) Original instance from CIFAR-10 dataset

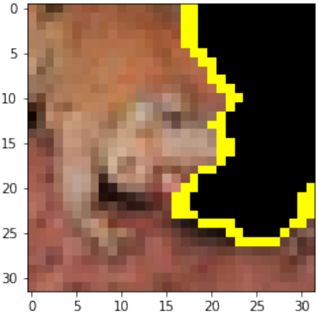
(b) Local explanation for misclassification as DEER

Figure: LIME explanations for CIFAR-10 dataset instance

Results: LIME Explanations for CIFAR-10



(a) Original instance from CIFAR-10 dataset



(b) Local explanation for correct classification

Figure: LIME explanations for CIFAR-10 dataset instance