2019年度アドバンス・トップエスイー

最先端工学ゼミ　個別ゼミⅠ成果発表
機械学習グループ

～　機械学習の説明可能性　～

伊田　侑起
工藤　淳真
永田　哲也

目次

- 説明技術の必要性
- 目的
- 説明技術とは
- ゼミで実施した手法の説明
- 取り組み内容(データ種類毎)
  - テーブルデータ
  - テキストデータ
  - 画像データ
- 結果/評価/考察
- Future Work

説明技術の必要性

1.利用者を含むステークホルダへの説明

背景：サービス提供者に説明責任が求められつつある
モデルを導入・拡大するには判断根拠を説明できることが重要
方法：機械学習モデルが判断のために特に重視して用いた情報

2.機械学習モデルの信頼性の向上

背景：機械学習モデルはブラックボックスになりやすく、安易に信頼できない
高精度な予測ほどモデルを読み解くことが難しくなる傾向
方法：判断内容にどの特徴量がどの程度寄与していたかを提示

3.機械学習モデルの品質の向上

背景：機械学習開発におけるテスト手法の未確立
方法：説明モデルを開発工程のテストツール、デバッグツールとして使用

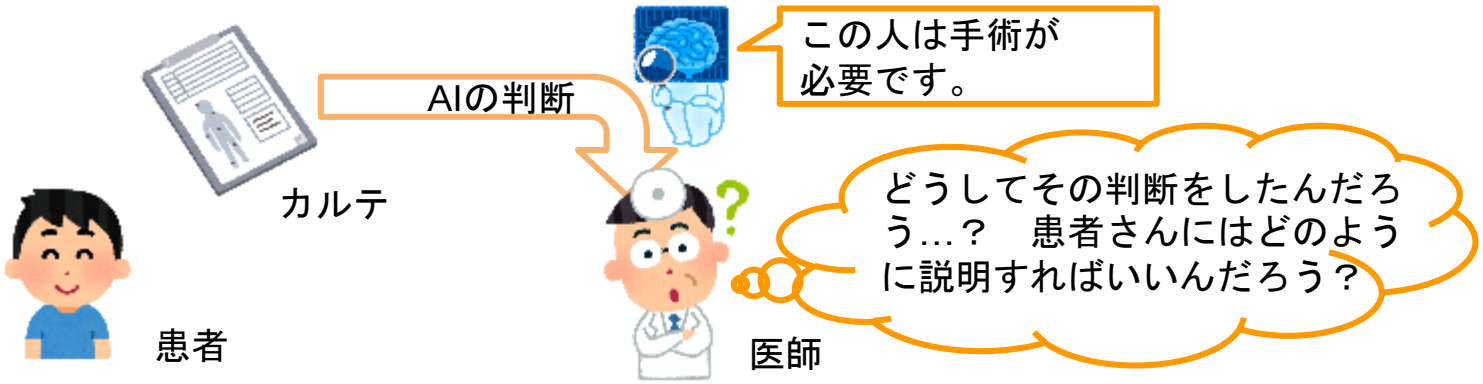
https://speakerdeck.com/line_developers/machine-learning-and-interpretability?slide=6
https://www.ai-gakkai.or.jp/my-bookmark_vol33-no3/

目的：ゼミで実施したい事

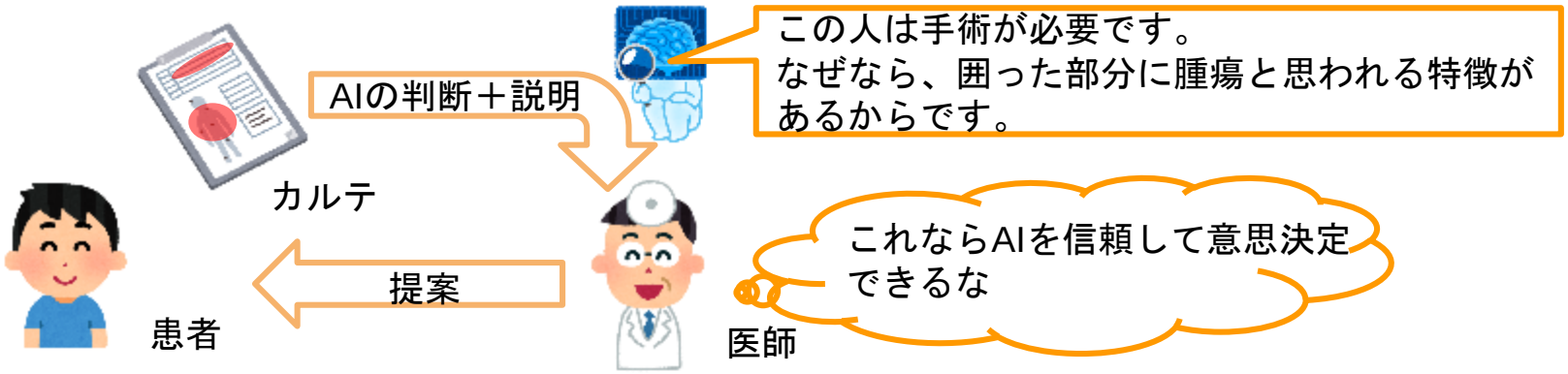
- 説明モデルを使用してみたい
  - ○3人中2人が説明モデルを使ったことがない
- データの種類・説明モデルごとに特徴を調べる
  - ○対象データの種類はテーブルデータ、テキストデータ、画像データ
  - ○説明モデルはLIME、SHAP、Grad-CAM、DefragModel
- 説明技術がどのように活用できるかを体験する
  - ○エンジニア向けか、運用向けか、どのタスクに使えるかなどを考える

説明技術とは？

- 機械学習のブラックボックス問題
  - 機械学習は高精度だが、判断の根拠・理由の説明をしてくれない
  - 機械学習による意思決定をした際の過誤の原因解明や、説明責任を果たせない

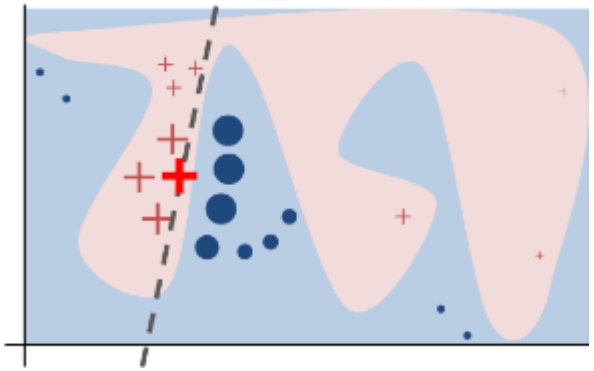


- 説明技術
  - 機械学習の予測結果、モデル自体に対して、**特徴量などの入力された情報を提示することで人間に理解可能な判断根拠の説明を行う手法**（**例では特徴量は診断書の数値や画像を想定**）
  - 入力データに対する予測の説明を**局所的説明**、学習モデル自体の説明を**大域的説明**という



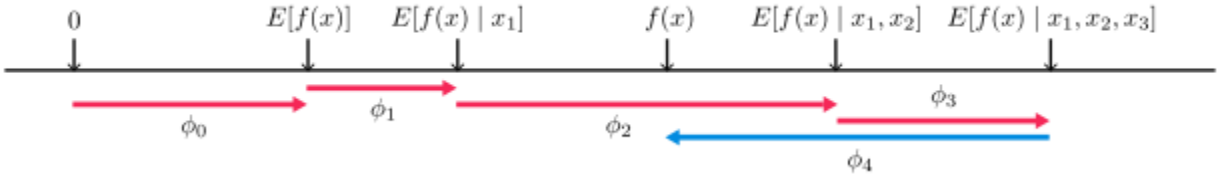
ゼミで実施した手法の説明

1. LIME
   - 予測モデル（確率モデル）f：R^d→R、解釈可能ベクトルR^e、入力したデータの周辺をサンプリングした解釈可能ベクトルとのペア、Z＝（x、x’）とする。
   - Zにおいてfを線形モデル　g：R^e→R、g（x’）＝wx’で線形近似を行い、wを特徴量寄与度として出力することで説明を行う手法。



最も大きい＋が予測データ、
＋が正例、●が不例、破線が線形近似結果

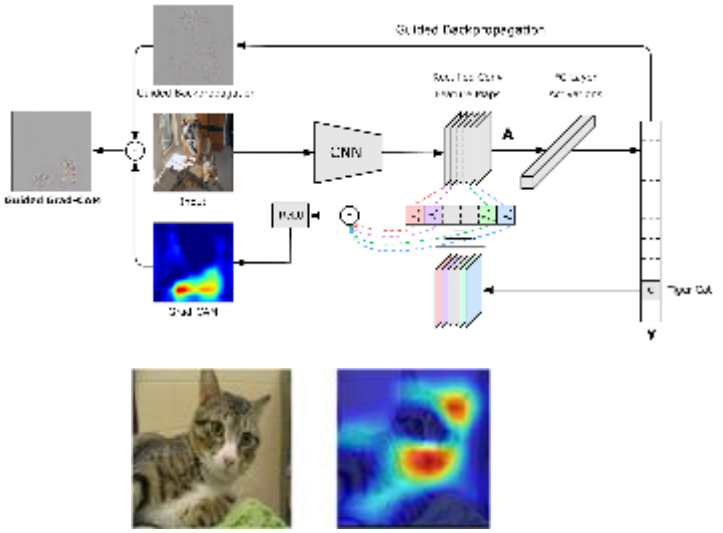
2. SHAP
   - LIMEでは線形近似＋統計値による解釈可能ベクトルの表現を行っているが、本当に確率に寄与するか疑問があった。
   - SHAPでは、特徴量をゲームのプレイヤー、予測確率を利得とみた協力ゲームと見て、特徴におけるシャープレイ値（全員が協力したときに得られた利得をプレイヤーに分配するようなプレイヤーの貢献度）を計算し、予測確率における特徴量の寄与を提示することで説明する手法。



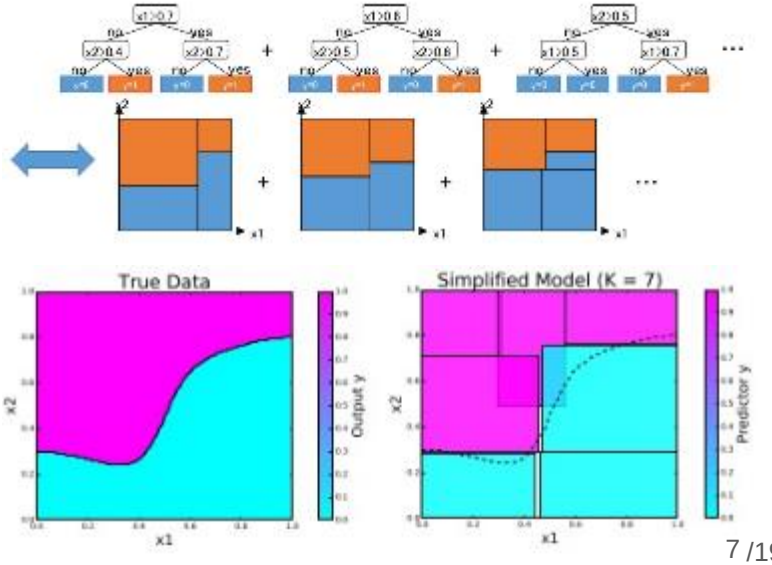
各特徴の寄与度（Φ_1,Φ_2,Φ_3,Φ_4）が計算され、予測確率f(x)が再現される。

ゼミで実施した手法の説明

3. Grad-CAM
   - CNNが分類に寄与した（と思われる）部分をカラーマップで表示する手法。
   - CNNの後の畳み込み層における勾配を平均して、重要度を計算することでヒートマップを求める。



4. DefragModel
   - ランダムフォレストをルールベースの亜種として考えることでアンサンブル学習をルールで表現することで説明する大域的説明手法。
   - 既存のモデルを決定木で近似するBorn Again Treeでは木が深すぎて解釈が困難になる問題を、最小のルール表現を推定することで解決を図った



取り組み内容：テキストデータ

- 利用したデータ：livedoorニュースコーパス
  - 対象文書がどのカテゴリに属するかを当てる問題
    - 単語数：前処理前　64,334　、前処理後 14,553
    - ラベル：家電チャンネル、トピックニュース、MOVIE ENTER etc…
    - 利用した機械学習モデル：RandomForestClassifier
      - 精度：81.75%
- データセットの前処理
  - テキストクリーニング
    - 括弧・全角空白の除去等
  - 分かち書き（Mecab）
    - 今回はテキストから名詞だけを抽出
  - 単語の正規化
    - 数字の変換、全角半角の統一
  - ストップワード除去
  - 単語のベクトル化（TfidfVectrozer）
- 利用した説明技術: LIME, SHAP, DefragModel

  85.3s　　2.4s　　1.4s
  - 速度　　：DefragModel <<< SHAP ≒ LIME
  - 適用容易性：DefragModel = SHAP < LIME
  - 視認性　　：DefragModel <<< SHAP ≒ LIME

取り組み内容：テキストデータ

1. LIME

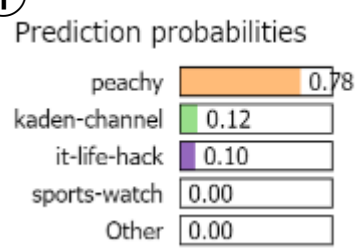
1テキストデータごとの

①各カテゴリに対する予測確立を表示

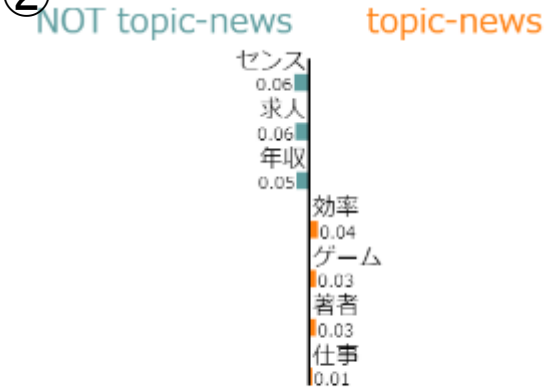
②topic-newsかtopic-news以外かの単語ごとの寄与率を表示

③ハイライトされた単語の色の濃さによってどの程度寄与しているかを視覚的に表示

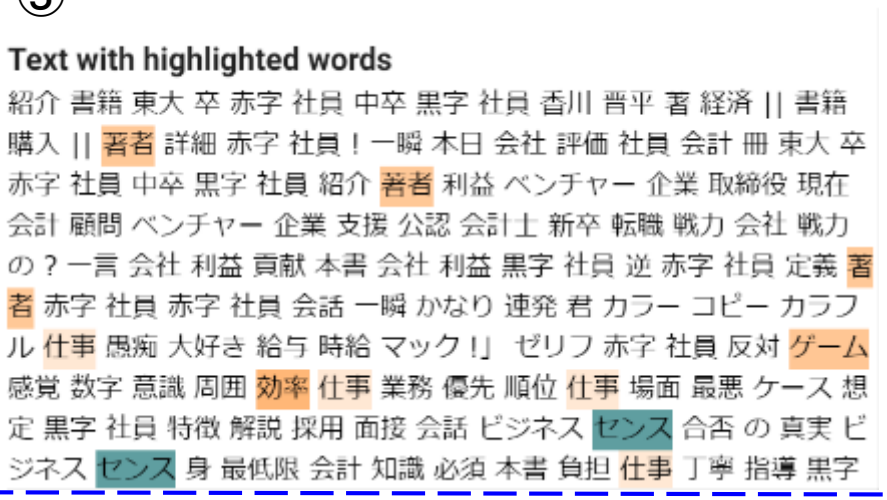
①



②



③



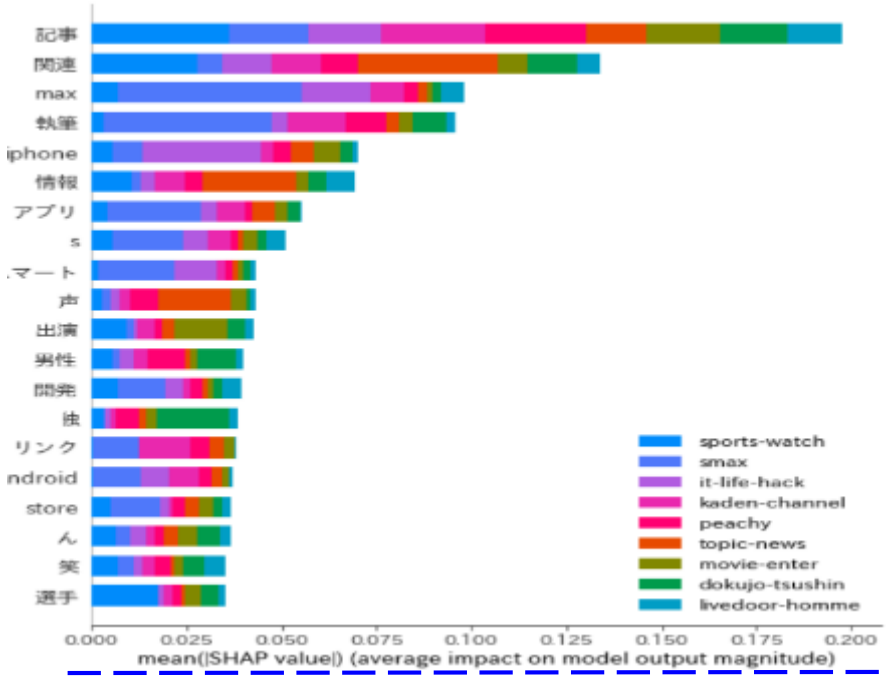
ていう くらい 野菜 が 届き ます 」 と 真美 さん （ 34 歳 ） は 困り 顔 だ 。 「 すでに リタイア し て いる 両親 は 家庭 菜園 が 日課 で 、 たまに 野菜 を 送っ て くれ ます 。 でも それ が 白菜 を 2 株 、 大根 3 本 、 ミョウガ 60 個 とか 、 保存 が 利く から と ジャガイモ や カボチャ を 売る はど （ 10 kg くらい ） 大量 な ん です よ 。 一人暮らし です から 到底 食べ きれる わけ も なく 、 結局 ダメ に する こと も ざ て 。 「 若い 人 は たくさん 食べる ！ 」 って 思っ てる ん です よ ね 。 友人 、 知人 、 同僚 に 配る に し て も 、 ちょっと 恥ずかしい し 。 何 度 言っ て も 、 言う こと 聞か ない ん です よ （ 泣 ） 」 修子 さん （ 38 歳 ） は 、 親 から 「 ちゃんと ご飯 を 食べ てる の ？ 」 と 聞か れる の が 嫌 だ という 。 「 私 、 ご飯 を たくさん 食べる ほう な ん です けど 、 あまり 太ら ない 体質 な ん です 。 だから こう い

前処理を行う前の文章で学習モデルを構築し、
LIMEに適用した結果が右

LIMEの出力結果より、テキストデータ前処理が不十分だとわかる
※今回の場合だと、学習モデルが副詞、助詞等を重要視しているとわかる
→デバッグプロセスへの適用

取り組み内容：テキストデータ

- SHAP
  - 単語がどの程度カテゴリに寄与しているかを表示
  - 複数テキストの単語寄与率を確認できる



- DefragModel
  - カテゴリ毎に単語ベクトルのルールを表示
  - Defrag Modelのエラー率が50%以上で、説明モデルの信頼度が低い

[Rule 3]
y = 6 when
じゃま < 0.09129
すり < 0.033222
ジャム < 0.01520
チャンス < 0.014
ビデオ < 0.07095
ブログ < 0.04502
プリント < 0.040
ボディ < 0.02311
ライフハック < 0.
勝 < 0.097256
塾い < 0.047106

[Rule 2]
y = 2 when
optimus < 0.041318
q < 0.030927
white < 0.021010
じゃま < 0.091292
ウィンドウ < 0.019134
チーム < 0.029482
ハイナー < 0.058840
ファン < 0.103458
フィット < 0.021347
プレゼント < 0.075927

各説明モデル特徴

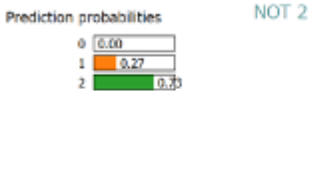
- LIME ：学習モデルの適用が容易であり、説明結果の視認性も高い
- SHAP：複数のテキストより、単語の寄与率を見たいときに最適
- DefragModel：説明モデルの学習に時間がかかる。また、説明結果を見ても学習モデルの説明やデバッグプロセスにも適用が難しい

取り組み内容：テーブルデータ

- 利用したデータ: **Wine Quality Data Set**
  - ワインの化学成分からワインの品質を当てられるかという問題
    - 特徴量: 化学成分(アルコール度数, pH,…)
    - ラベル: 品質(3~8の整数値)
  - このデータに回帰と分類(3~4を低, 5~6を中等, 7以上を高と離散化)問題としてアプローチ
  - 利用した機械学習手法: ランダムフォレスト分類・回帰
- 利用した説明技術: LIME, SHAP, DefragModel
  - 速度: DefragModel < LIME < SHAP

    モデル作成+1データの説明　30min↑　1.6s　1.5s
  - 汎用性: DefragModel < LIME ≒ SHAP
  - 視認性:
    - 分類: DefragModel < SHAP < LIME (LIMEはアルゴリズム的に解釈ベクトル作成を含むので)
    - 回帰: DefragModel < LIME < SHAP (予測値を再現する特徴量内訳がでるので)

LIME



SHAP

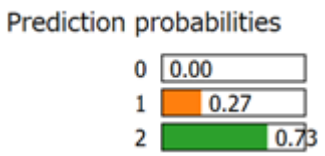
DefragModel

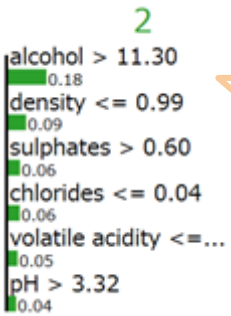
```
[Rule  1]
y = 1 when
        fixed acidity < 10.050000
        volatile acidity < 1.047500
        citric acid < 0.870000
        0.022000 <= chlorides < 0.205000
        free sulfur dioxide >= 6.000000
        total sulfur dioxide >= 29.000000
        0.991190 <= density < 0.997600
```

取り組み内容：テーブルデータ

- LIME vs SHAP

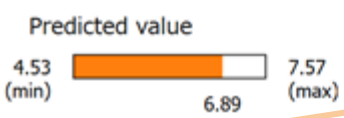
分類における比較

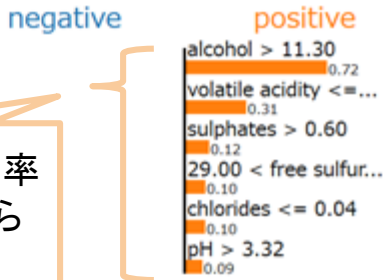




LIMEでは単純に特徴量を出すのではなく解釈ベクトルを設計していて、alchol>11.3なので2を予測したという説明になっている

回帰における比較





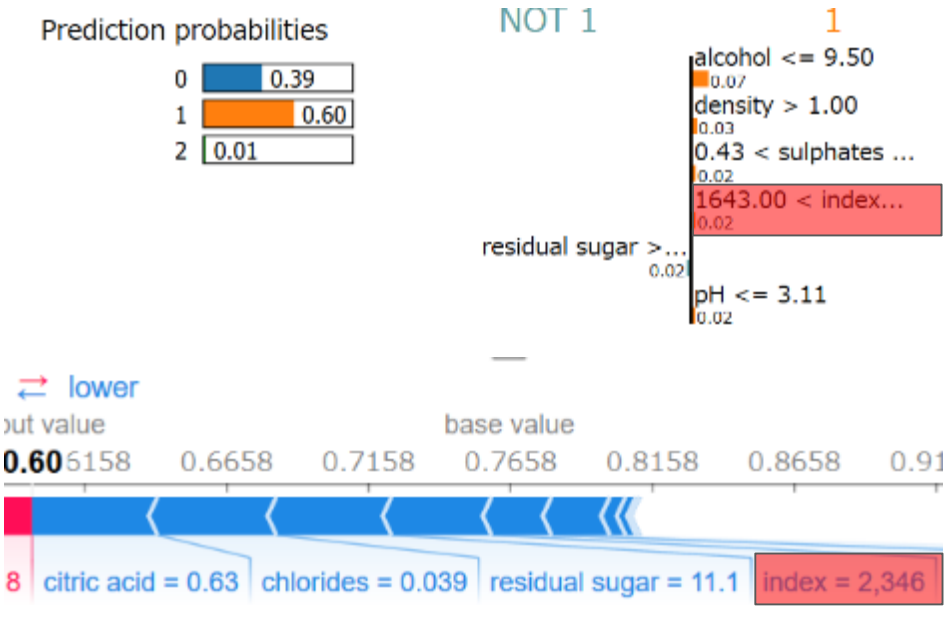
LIMEでは特徴量寄与率の合計が予測値とならない

SHAPでは予測した値に寄与した特徴量の内訳が出る

回帰ではSHAP, 分類ではLIMEかSHAP, 特に、顧客から解釈ベクトルを設計するドメイン知識を得られる場合はLIMEを選択するのが良いだろう。

取り組み内容：テーブルデータ

- 間違った特徴を探す
  - 間違った特徴index番号などを入れて、どれだけ間違った特徴に左右されるかをみる。



LIMEでもSHAPでも、誤ったデータの判断根拠として誤った特徴を提示した

開発時においてデータ分析者がモデル作成、デバッグプロセスを回す上で実用可能

- DefragTreeの問題点
  - DefragTreeは大域的な説明手法として解釈性も優れる手法であるが以下の問題点が分かった
    - ラベル数が5を超えや2000件のデータでも30mを超える計算時間を要する。
    - 得られるルールが全てのラベルを網羅しない
    - 回帰の場合特定の値しか出ないので精度が著しく悪い

```
Optimal Model >> Seed  19, TrainingError = 0.67
[Rule  1]
y = 5.553875 when
        fixed acidity >= 5.000000
        volatile acidity >= 0.110000
        residual sugar >= 1.300000
        chlorides >= 0.028000
        free sulfur dioxide < 133.250000
        density >= 0.992860
        sulphates >= 0.285000
```

回帰のDefragTree

取り組み内容：画像データ

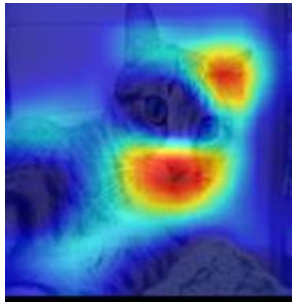
- 利用したデータ: Dogs-vs-cats-redux-kernels-edition
  - 犬と猫の画像をうまく分類できるかという問題

- 利用した学習モデル: 犬猫分類器
  - **VGG16からFineTuning**・・・特徴量抽出部分はそのまま分類を学習
    (1000クラス分類→2クラス分類)

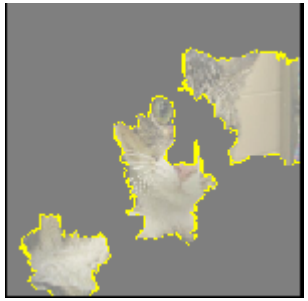
- 利用した説明技術: **Grad-CAM**, **LIME**, **SHAP**

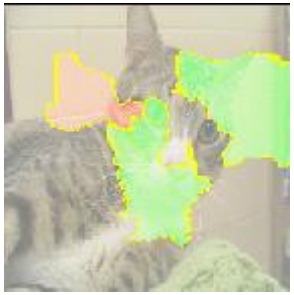
- ➢ 判定影響方向有：LIME,SHAP
- ➢ 速度(遅→速)：LIME < SHAP < Grad-CAM
  画像30枚 480秒, 120秒,　20秒 ← GK210GL [Tesla K80]
- ➢ 適用容易性(難→易)： Grad-CAM,SHAP << LIME
- ➢ 視認性(→良)：LIME,SHAP < Grad-CAM

■ Grad-CAM

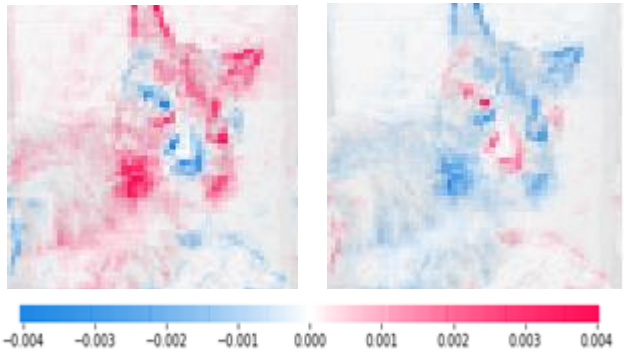


■ LIME





■ SHAP



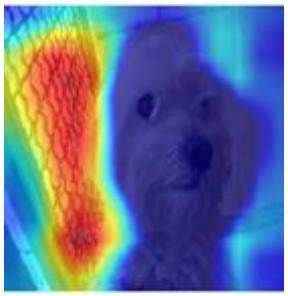
取り組み内容：画像データ (判定NG 特徴箇所間違え)

- モデル構築条件 (未学習気味)
  - 学習1000枚/検証1000枚
- 入力画像：犬
  判定：52.5%猫
  (自信なく誤判定)

説明技術：
背景を特徴として判断
→モデルとして悪い!

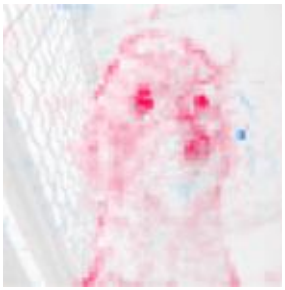
対策
案1:学習画像を増やす
案2:背景をマスクして学習

■ Grad-CAM



■ LIME

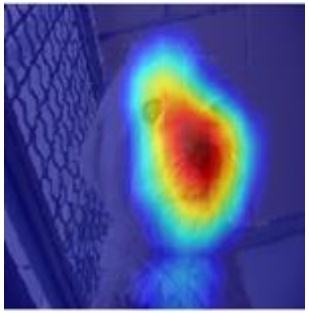
■ SHAP

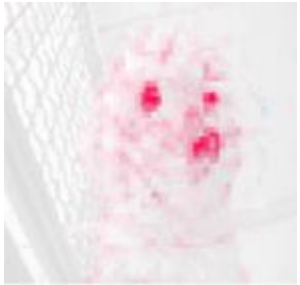


案1
実行

- モデル構築条件 (検証精度：95.6%)
  - 学習12000枚/検証1000枚

判定：75.9%犬





・犬の特徴箇所で判定(推測)
この画像では改善した

・顧客への説明に有用
Grad-CAM (LIMEも優)
・デバッグに有用
Grad-CAM ,LIME
※ 説明技術間の差異有

SHAPでは正規化画像データに対応出来なかった
(Grad-CAMは正規化画像のみ。LIMEはどちらもOK)
正規化：8bit画像を255で割る

背景で判定しても
正解していればいいか？
（金網、首輪など）

取り組み内容：画像データ (判定NG 特徴判定間違え)

- 入力画像：犬
  判定：84.6%猫
  (自信有→誤判定)

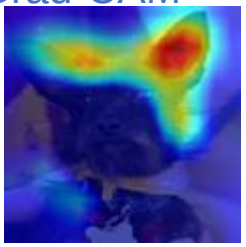
説明技術：
　耳の形から猫と判断 (推測)

対策

案1:猫っぽい耳の犬を学習

案2:顧客に判断を依頼
(この耳なら仕方がない、
耳以外は犬と説明できるなど)

■ Grad-CAM



■ LIME

■ SHAP

- 顧客に説明する際の説明技術
  - 3つとも視認性に優れている
  - ケースによって使い分けたい
    - 視認性と速さを重視：Grad-CAM
    - 適応容易性,汎用性を重視:LIME
    - 画像全体の影響を重視:SHAP
      (ケースによっては強みがある)

- 運用での説明技術
  - 説明技術も間違えることもあるが
    参考になり、非常に有用
  - 全ての画像に適用するのは非現実的

  → 自信のないものから探す

  → 自信を持って間違えるものはどのようにして探すか？

各説明モデル特徴

- Grad-CAM： 視認性と速さに優れている
- LIME ：学習モデルの適用が容易であり、視認性も高い
  速度を気にしないならこの技術でまず取組みたい
- SHAP：画像全体の影響を知るには最適

結果/ 評価/ 考察

- 説明モデルを使用してみたい
- データの種類・説明モデルごとに特徴を調べる
- 説明技術がどのように活用できるかを体験する

◎：比較して大いに優れる
〇：優れる
△：劣る

データ種類	LIME	SHAP	Grad-CAM	DeflagModel
テキスト	総合評価：◎ 適用容易性：◎ 速度：◎ 視認性：〇	総合評価：〇 適用容易性：〇 速度：〇 視認性：〇	今回未実施	総合評価：△ 適用容易性：△ 速度：△ 視認性：△
テーブル	総合評価：◎ 適用容易性：◎ 速度：◎ 視認性：〇	総合評価：◎ 適用容易性：〇 速度：◎ 視認性：〇	×	総合評価：△ 適用容易性：△ 速度：× 視認性：〇
画像	総合評価：◎ 適用容易性：◎ 速度：△ 視認性：〇	総合評価：〇 適用容易性：△ 速度：〇 視認性：〇	総合評価：〇 適用容易性：〇 速度：◎ 視認性：◎	×

※　総合評価はケースによって異なる

結果/ 評価/ 考察

- 説明モデルを使用してみたい
- データの種類・説明モデルごとに特徴を調べる
- 説明技術がどのように活用できるかを体験する
  - 開発への適用
    - 😊 デバッグプロセスに利用可能
    - 😕 モデルがなぜ間違っているかを示すデータを見つけるのが困難であり、多くのデータを見る試行錯誤が必要
      （例1：猫の画像を猫と判断しているが説明では背景を見ているケース
      例2：猫の画像を犬と判定しているが耳を見ているようなケース）
    - 😕 モデルの間違いを示すが、次にどのような改善をすべきかはデータ分析者が判断し、経験や能力に依存する
    - 😣 複数の説明技術を適用して異なる結果を出す事がある
      (説明技術の信頼性)
  - 運用への適用
    - 😊 理解不能なブラックボックスを理解可能にできる
    - 😕 様々な説明可能手法のうち、どの手法が顧客が満足する説明技術なのかを評価する手段がない

Future work

- 達成した事
  - 説明モデルを使用してみる
  - データの種類・説明モデルごとに特徴を調べる
  - 説明技術がどのように活用できるかを体験する

  （ゼミ当初目的）
  - テーブルデータでは様々なケース(分類/回帰/知識ドメインの有無)での説明技術優劣まで踏み込んだ検証
- 今後取組みたい事
  - データ分析コンペ等で説明技術を適用し、デバッグに使えるかを実践する
  - 最新技術の動向調査、実施
  - 説明技術のアルゴリズムを深堀した検証
  - 説明技術の信頼性

参考

[1] “機械学習と解釈可能性 / Machine Learning and Interpretability” https://speakerdeck.com/line_developers/machine-learning-and-interpretability?slide=6

[2] ”【記事更新】私のブックマーク「機械学習における解釈性（Interpretability in Machine Learning” https://www.ai-gakkai.or.jp/my-bookmark_vol33-no3/

[3] “最先端ソフトウェア工学ゼミ[13期, 2018年度] 機械学習ゼミ(第2期)” https://www.topse.jp/images/機械学習ゼミ.pdf

[4] ”Lime: Explaining the predictions of any machine learning classifier” https://github.com/marcotcr/lime

[5] “SHAP (SHapley Additive exPlanations)” https://github.com/slundberg/shap

[6] ”Grad-CAM: Gradient-weighted Class Activation Mapping” http://gradcam.cloudcv.org/

[7] “defragTrees” https://github.com/sato9hara/defragTrees

[8] “livedoor ニュースコーパス” https://www.rondhuit.com/download.html#ldcc

[9] “Wine Quality Data Set” https://archive.ics.uci.edu/ml/datasets/wine+quality

[10] “Dogs vs. Cats Redux: Kernels Edition” https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition/rules

[11] “Very Deep Convolutional Networks for Large-Scale Image Recognition” http://www.robots.ox.ac.uk/~vgg/research/very_deep/

[12] “VGG16のFine-tuningによる犬猫認識” http://aidiary.hatenablog.com/entry/20170110/1484057655

[13] ”The (Un)reliability of saliency methods” https://arxiv.org/abs/1711.00867

以降 補足資料

ご清聴ありがとうございました

補足資料　説明技術の信頼性

説明モデルが正しいか？根拠を問われた場合

→　案①　複数説明モデルを適用する

・同じ結果を示した場合

信頼性がありそうだと言えそう

・全く異なる結果が得られた場合

どの説明モデルが正しいかはアルゴリズムを理解して、
アウトプットが意味する事からどの部分を信頼するかは
データ分析者のスキルに依存する

→　案②　説明技術の説明技術で説明する？

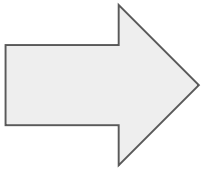
・説明技術を間違えさせる方法も示されているので、
必ず正解を出す説明技術は困難

”The (Un)reliability of saliency methods” https://arxiv.org/abs/1711.00867

補足 SHAP

協力ゲームのシャプレーイ値

A	B	C	利得
1	0	0	40
0	1	0	20
...			
1	1	1	100



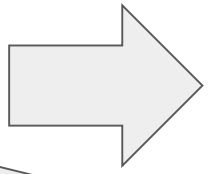
A	52.5
B	32.5
C	15

i番目のシャプレーイ値の算出式

$$\phi_i = \sum_{S \subset N \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M!} [f_x(S \cup \{i\}) - f_x(S)]$$

確率モデルを協力ゲームモデルとして見る

特徴量1	特徴量2	特徴量3	f(x)
x1	0	0	0.51
0	x2	0	0.6
...			
x1	x2	x3	0.72



A	0.02
B	0.1
C	0.1

協力してない特徴量は
欠損していると見る