第2回
やまぐち
高校生データサイエンティスト育成講座

[confidential] Copyright© 2021 SIGNATE Inc.

モデルの評価方法

- モデルの評価はとても重要
- 予測問題に応じて評価方法が異なる

<E.g.>

- 豆腐の需要予測
  - 予測値と実測値の誤差がどのくらいあるか？
- ネット広告のクリック予測
  - 予測がどれくらい信用できるか？
- 検索エンジン
  - ユーザが見たい記事が検索できているか？

[confidential] Copyright© 2021 SIGNATE Inc.

モデルの評価方法

- モデルの評価はとても重要
- 予測問題に応じて評価方法が異なる

<E.g.>

- 豆腐の需要予測
  - 予測値と実測値の誤差がどのくらいあるか？
- ネット広告のクリック予測
  - 予測がどれくらい信用できるか？
- 検索エンジン
  - ユーザが見たい記事が検索できているか？

評価関数

[confidential] Copyright© 2021 SIGNATE Inc.

評価関数とは？

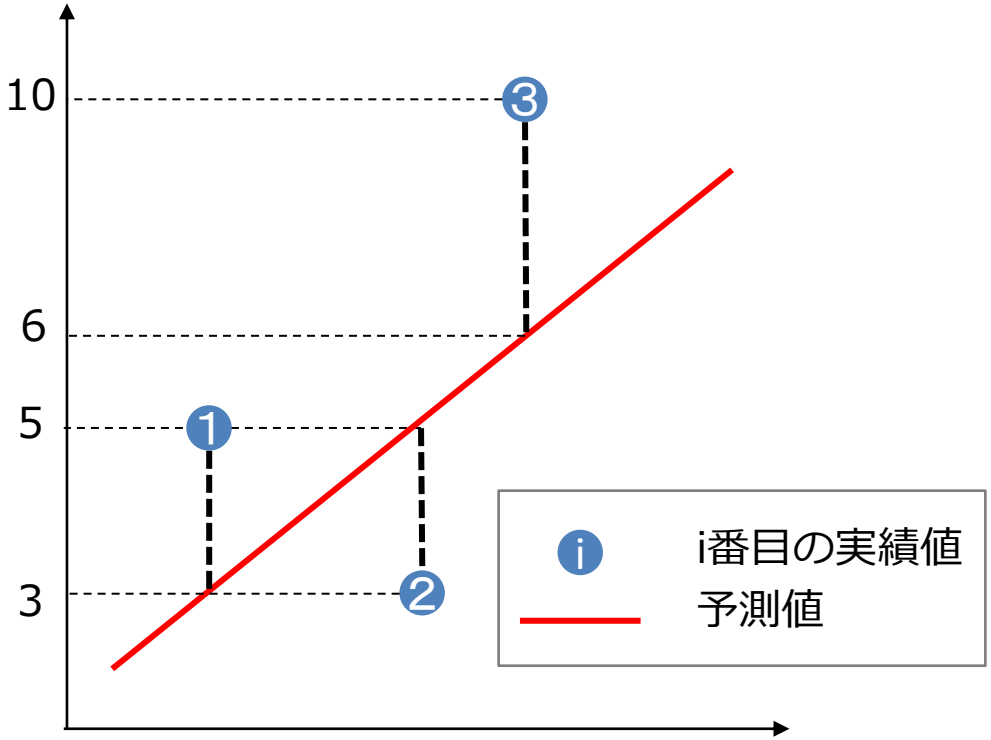
- モデルの予測精度を評価する数式
  - 今回はRoot Mean Squared Error(RMSE)
  - RMSEは誤差を表す指標の為、少ないほど良い

$$RMSE = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(y_i - p_i)^2}$$

N:予測対象数
y_i:i番目の実績値
p_i:i番目の予測値

[confidential] Copyright© 2021 SIGNATE Inc.

RMSEの計算の例



	1	2	3
実績値	5	3	10
予測値	3	5	6
実績値-予測値	2	-2	4
↑の二乗	4	4	16

$$RMSE = \sqrt{\frac{1}{3}(4+4+16)} = \sqrt{8} \fallingdotseq \mathbf{2.828}$$

[confidential] Copyright© 2021 SIGNATE Inc.

色々な評価関数

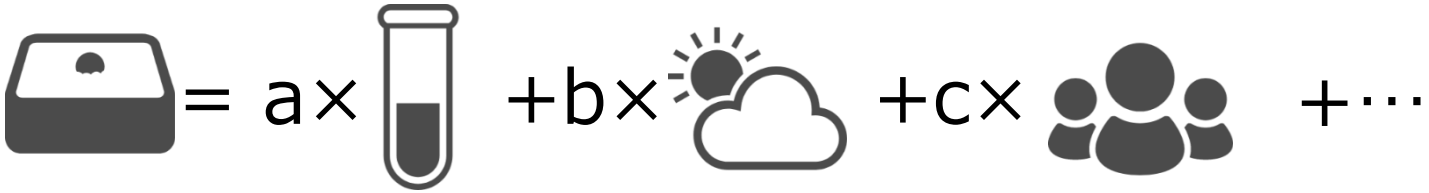
評価関数	予測対象	値域	見方
AUC	分類精度を測る e.g) 医療診断	0～1	値が大きいほど良い
LogLoss	分類精度を測る e.g) 画像分類	0～∞	値が小さいほど良い
Accuracy	分類精度を測る e.g) 画像分類	0～1	値が大きいほど良い
Precision	正確性を測る e.g) 検索エンジン	0～1	値が大きいほど良い
Recall	カバー率を測る e.g) 検索エンジン	0～1	値が大きいほど良い
MAE	誤差を測る e.g) 需要予測	0～∞	値が小さいほど良い
MAP@N	検出精度を測る e.g) 推薦エンジン	0～1	値が大きいほど良い
nDCG	ランキング精度を測る e.g) 推薦エンジン	0～1	値が大きいほど良い

[confidential] Copyright© 2021 SIGNATE Inc.

重回帰とは？

- ２つ以上の説明変数を使った回帰モデル
  - 予測する為の手掛かりをたくさん使う

$$y = ax_1 + bx_2 + cx_3 + \cdots$$



お弁当の売上　気温　天気　来店数

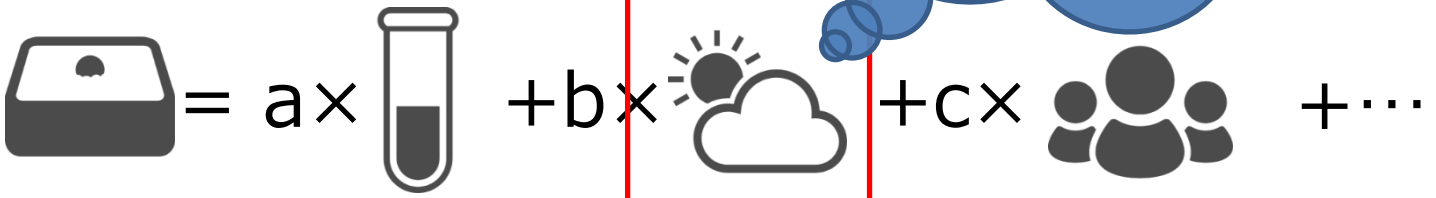
[confidential] Copyright© 2021 SIGNATE Inc.

重回帰とは？

- ２つ以上の説明変数を使った回帰モデル
  - 予測する為の手掛かりをたくさん使う

$$y = ax_1 + bx_2 + cx_3 + \cdots$$

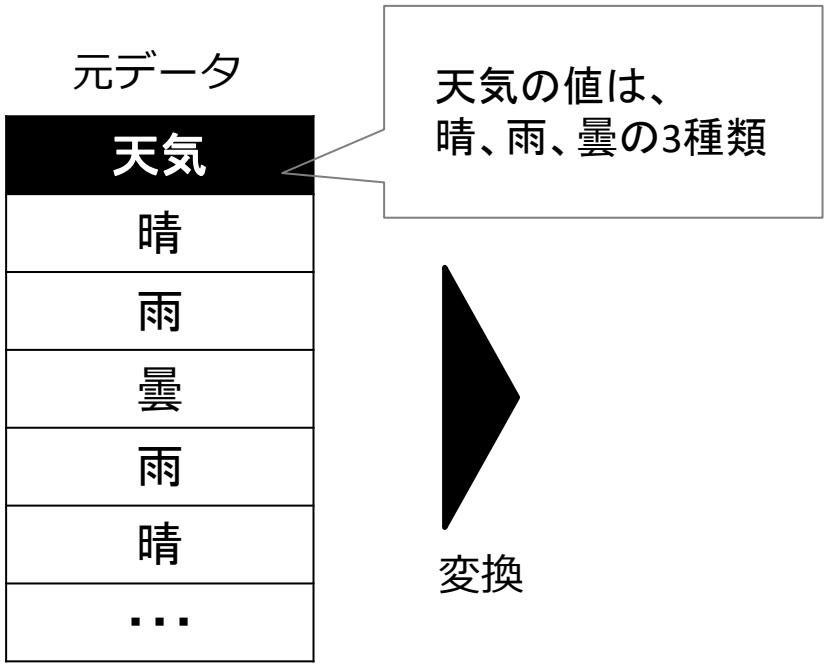
数値じゃないのにどうするの？？？



[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）



[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

変換

1-of-K表現

晴	雨	曇
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

変換

1-of-K表現

晴	雨	曇
1	0	0
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

1-of-K表現

晴	雨	曇
1	0	0
0	1	0
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

1-of-K表現

晴	雨	曇
1	0	0
0	1	0
0	0	1
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

1-of-K表現

晴	雨	曇
1	0	0
0	1	0
0	0	1
0	1	0
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

1-of-K表現

晴	雨	曇
1	0	0
0	1	0
0	0	1
0	1	0
1	0	0
…		

[confidential] Copyright© 2021 SIGNATE Inc.

ダミー変数

- 質的データを数値データへ変換する
  - 1-of-K表現（one hot encoder）

元データ

天気
晴
雨
曇
雨
晴
…

1-of-K表現

晴	雨	曇
1	0	0
0	1	0
0	0	1
0	1	0
1	0	0
…		

[confidential] Copyright© 2021 SIGNATE Inc.

予測モデルの予測精度を上げるには？

- 特徴量＝説明変数
  - 機械学習の領域で使われる
- 予測精度を上げる為には？

①特徴量を作る

②特徴量を選ぶ

[confidential] Copyright© 2021 SIGNATE Inc.

特徴量の作成

- 与えられたデータや外部データを加工し、予測の手掛かりとなりそうな新たな特徴を作ること

例１）基本統計量を作る

日付	店舗A	店舗B
4/1	100	50
4/2	60	40
4/3	70	50
4/4	140	90

AとBの平均
75
50
60
115

例２）データを集約する

Id	name	age
001	山田	23
002	鈴木	31
003	斉藤	18
004	藤井	29

年齢層
20代
30代
10代
20代

[confidential] Copyright© 2021 SIGNATE Inc.

特徴量の選択

- 特徴量が多すぎるのも良くない
  - 過学習のリスクがある

- 数ある特徴量から重要なもののみを選択することが大切
  - 単変量解析
    - 目的変数と各説明変数を1対1で確認し、取捨選択
    - E.g.) 分散分析など
  - モデルベース選択
    - モデルにとっての各変数の重要度を算出し、取捨選択
    - E.g.) ツリー系機械学習手法の重要度をみる
  - 反復選択
    - 特徴量を増減させながらモデルを生成し、良い特徴量を探索する
    - E.g.) ステップワイズ法など

[confidential] Copyright© 2021 SIGNATE Inc.

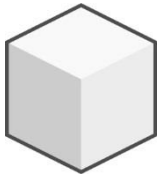
「特徴量を作ってみよう」の前に…

- 特徴量を作る為にはデータ加工が必要
- データ加工に便利な２つの関数
  - split関数
    - 文字列を分割する関数
  - apply関数
    - データの各値に数式や関数を適用する関数

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数

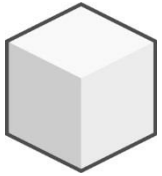


.　split(”区切り文字”)

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

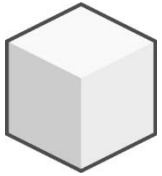
<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

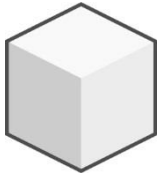
⇒colors = terms.split(“,”)

カンマ

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

⇒colors = terms.split(“,”)

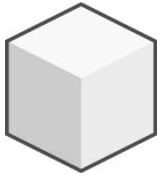
★colorsの中身を取り出したい

[“red”, “blue”, “green”, “yellow”]

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

⇒colors = terms.split(“,”)

★colorsの中身を取り出したい

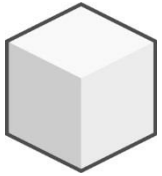
[“red”, “blue”, “green”, “yellow”]

0 1 2 3

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

⇒colors = terms.split(“,”)

★colorsの中身を取り出したい

[“red”, “blue”, “green”, “yellow”]

0 1 2 3

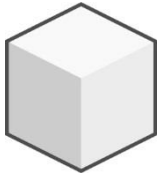
redを取り出したい

yellowを取り出したい

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

⇒colors = terms.split(“,”)

★colorsの中身を取り出したい

[“red”, “blue”, “green”, “yellow”]

0 1 2 3

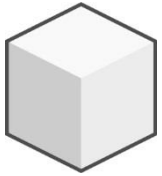
redを取り出したい⇒colors[**0**]

yellowを取り出したい

[confidential] Copyright© 2021 SIGNATE Inc.

split関数

- 文字列を指定した文字で分割する関数



. split(”区切り文字”)

<E.g>

terms = “red,blue,green,yellow”を**カンマ**で区切り、結果を変数colorsに代入したい

⇒colors = terms.split(“,”)

★colorsの中身を取り出したい

[“red”, “blue”, “green”, “yellow”]

0 1 2 3

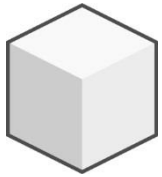
redを取り出したい⇒colors[**0**]

yellowを取り出したい⇒colors[**3**]

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

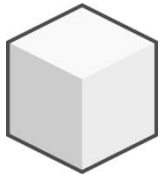


.　apply（lambda x : 〇〇）

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）



.　apply（lambda x : ○○）

数式 or 関数

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

dat[“value”].apply(lambda **x** : x*2+1)

dat =

Index	value
1	4
2	7
3	3
4	8

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

4*2+1

dat =

Index	value
1	4
2	7
3	3
4	8

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

4*2+1

9

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

dat[“value”].apply(lambda **x** : x*2+1)

dat =

Index	value
1	4
2	7
3	3
4	8

9

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

7*2+1

dat =

Index	value
1	4
2	7
3	3
4	8

9

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

7*2+1

9
15

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

9
15

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

3*2+1

9
15

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

3*2+1

9
15
7

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

9
15
7

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

8*2+1

dat =

Index	value
1	4
2	7
3	3
4	8

9
15
7

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

8*2+1

9
15
7
17

[confidential] Copyright© 2021 SIGNATE Inc.

apply関数

- 各値に数式や関数を適用する為の関数（pandas）

<E.g> datのvalueの各値を2倍して1を足したい

```
dat["value"].apply(lambda x : x*2+1)
```

dat =

Index	value
1	4
2	7
3	3
4	8

各値に「x*2+1」という
数式が適用された

9
15
7
17

[confidential] Copyright© 2021 SIGNATE Inc.