

予測問題: 発展 機械学習

川田恵介
東京大学
keisukekawata@iss.u-tokyo.ac.jp

2025-11-11

1 今後の予定

1.1 ここまで

- OLS/LASSO (線型モデル) を用いた
 - ▶ 予測分析: X (物件の属性) の情報から Y (取引価格) を推測するモデルを推定
 - ▶ 比較分析: 同じような X を持つ物件について、 D (改築済み/未改築) と Y (取引価格) の関係性を推定

1.2 課題

- 線型モデルを当てはめることが難しい母集団も存在
 - ▶ 予測精度の悪化、ミスリードな推定結果
- Y と D の関係性は、 X に依存
- より柔軟なアプローチが必要

1.3 実例

```
library(tidyverse)

data <- read_csv("Public/example.csv")

Y <- data$Price

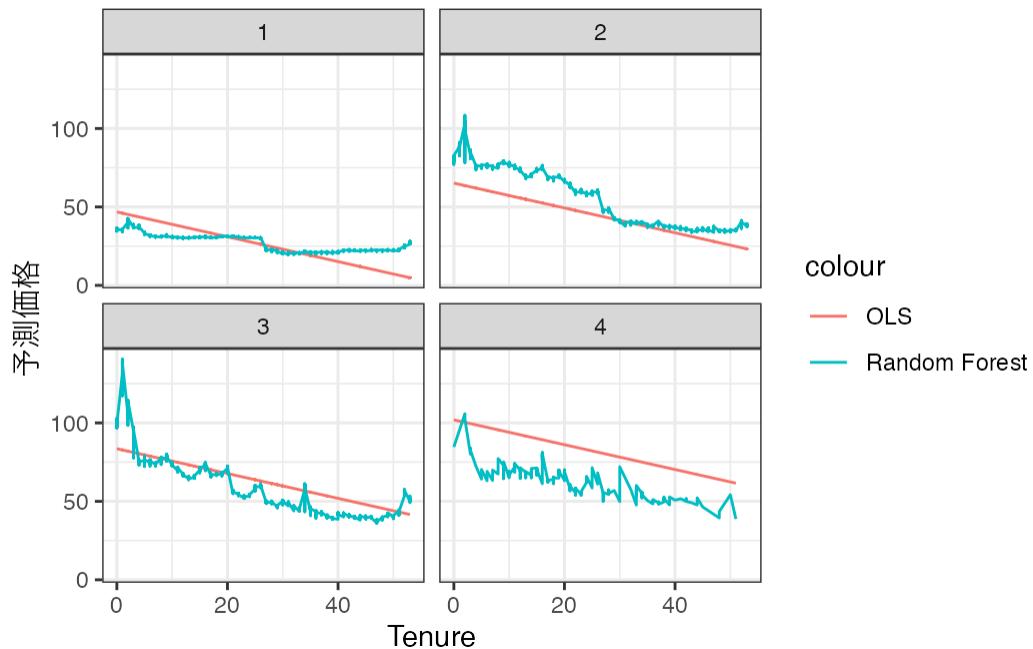
D <- data$year_2024

X <- model.matrix(
  ~ 0 + Tenure + RoomNumber,
  data
)
```

1.4 実例: 予測モデル推定

```
model_Y <- ranger::ranger(  
  y = Y,  
  x = X  
) # Random Forest
```

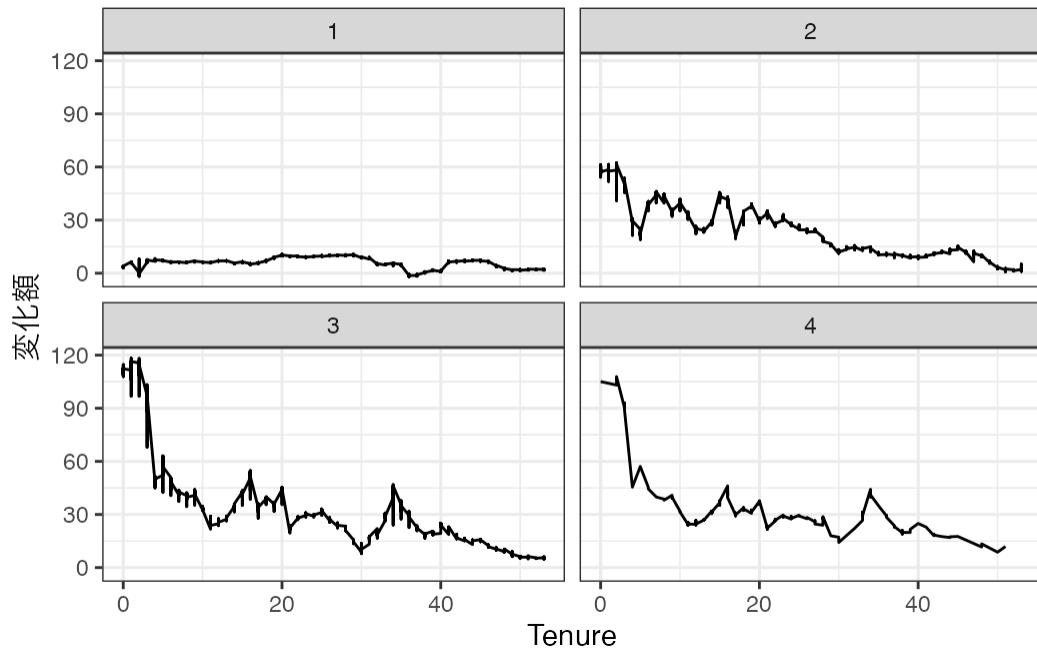
1.5 実例: 價格予測モデル



1.6 実例: 2024 VS 2021

```
model_Tau <- grf::causal_forest(  
  X = X,  
  Y = Y,  
  W = D  
) # Causal Forest
```

1.7 実例: 2024 VS 2021



1.8 RoadMap

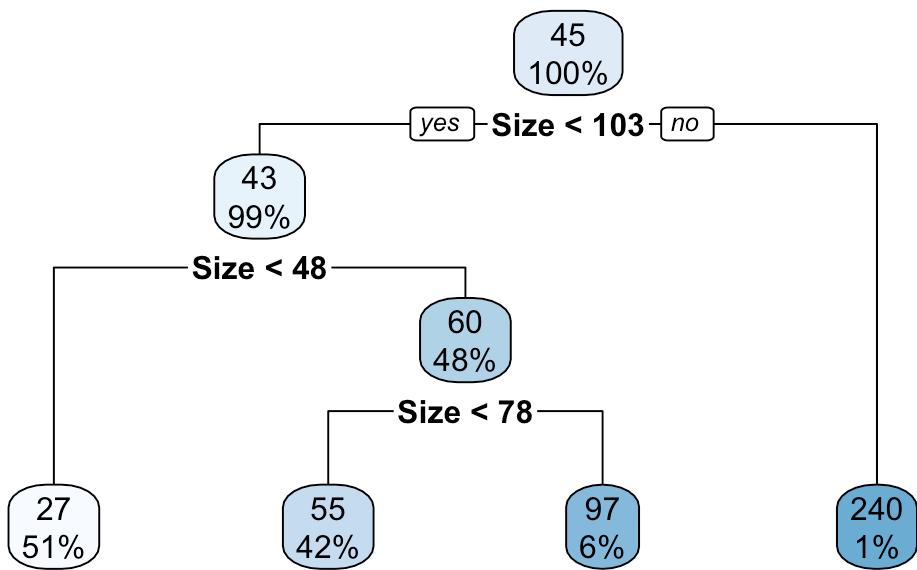
- ・線型モデルを補うために、回帰木モデルを改善する
 - Random Forest
- ・LASSO 以外も活用できる柔軟な比較分析の方法を紹介
 - Double/debiased machine learning

2 回帰木モデルの利点と弱点

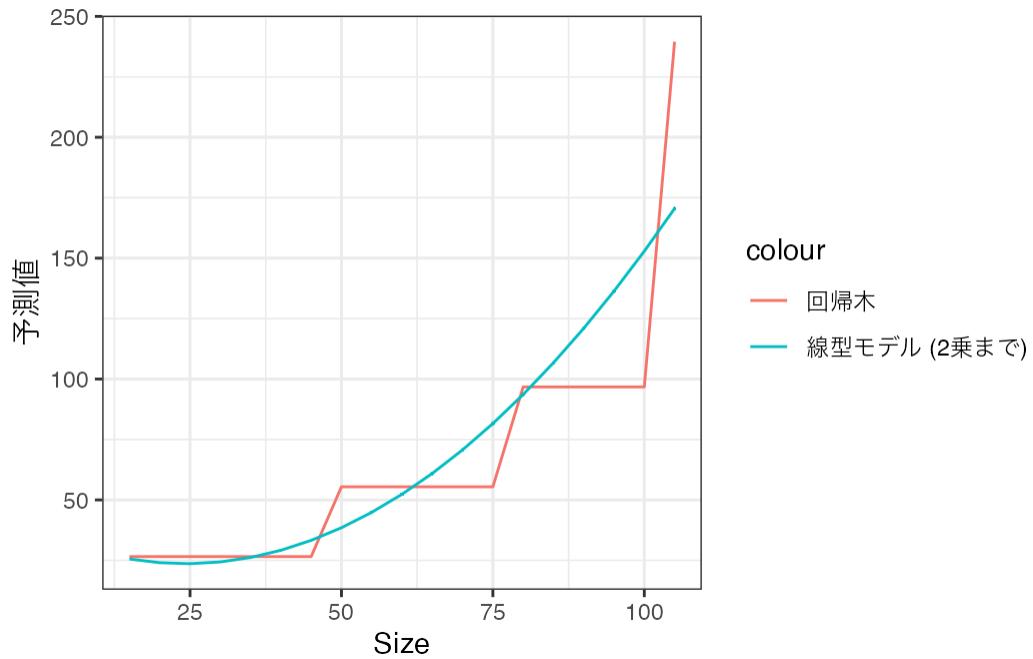
2.1 回帰木モデルの復習

- ・サブグループの平均値を予測値とする
 - サブグループは、データへの当てはまりを改善するように決定する

```
model <- rpart::rpart(Price ~ Size, data)
rpart.plot::rpart.plot(model)
```



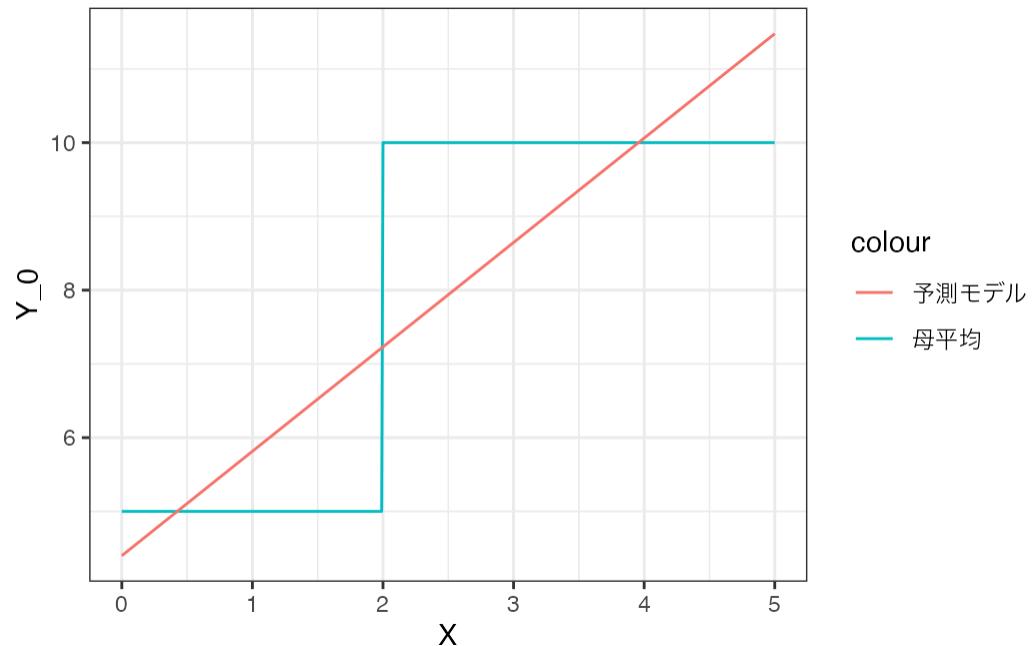
2.2 線型 VS 回帰木



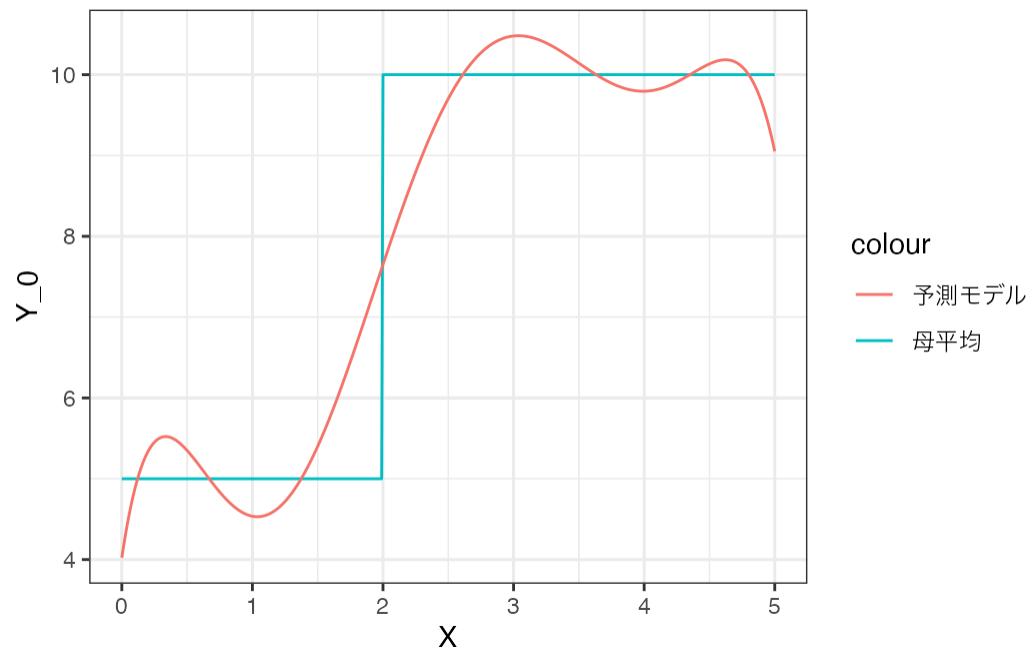
2.3 回帰木の利点

- ・線型モデルに比べて、 Y の母平均が”急変”する母集団に当てはめることが容易

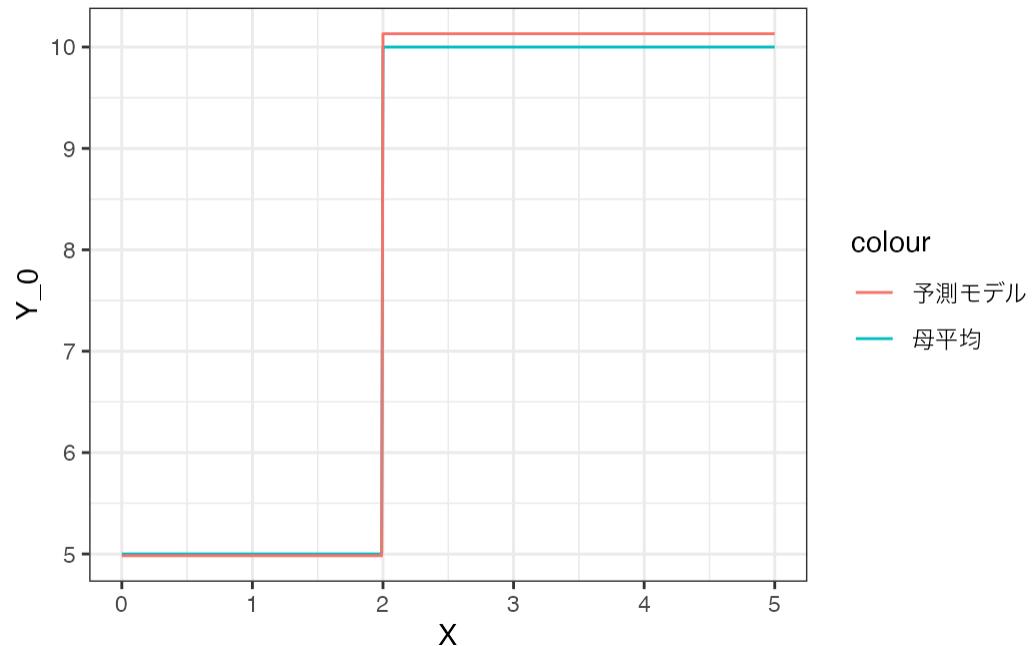
2.4 例: $Y \sim X$



2.5 例: $Y \sim X + X^2 + \dots + X^6$



2.6 例: 回帰木



2.7 弱点

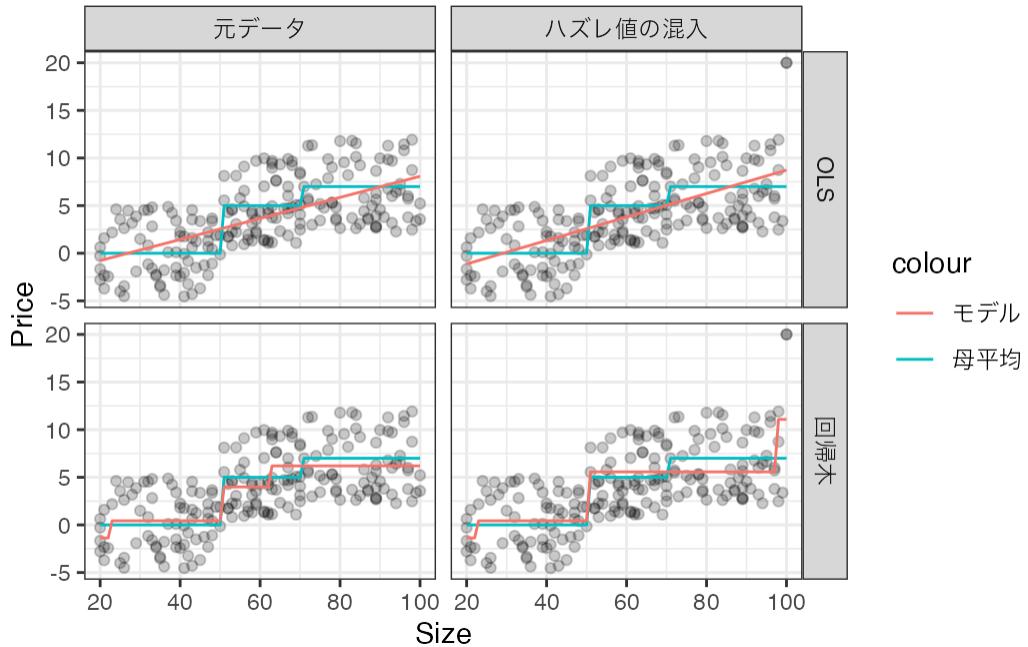
- データからの推定結果は、データの偏りの影響を受ける
 - ハズレ値の影響を受ける
- OLS や LASSO に比べて、回帰木はハズレ値の影響を受けやすい
 - モデル集計により解決できる

3 モデル集計

3.1 集計による解決

- データへの、平均値から極端に乖離した事例やその組み合わせの”混入”は、推定結果に大きな影響を与える
- モデルの単純化も選択肢だが、回帰木については不十分な場合が多い

3.2 数値例 (200 事例)



3.3 解決策

- 伝統的なアプローチ: “ハズレ値”を人間が除外
 - 採用するのであれば、“細心の注意”が必要
 - (議論はあるが)、極力避けた方が良い

3.4 集計による解決

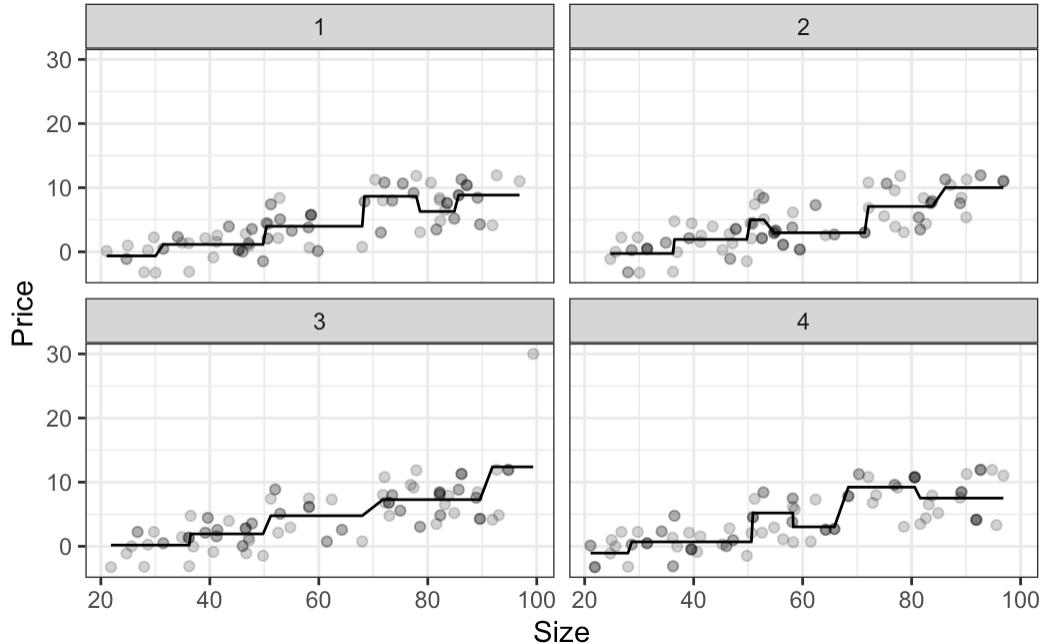
- モデルの集計
 - “異なる”データを用いた予測モデルの集計値(平均値)を最終予測とする
 - 特定のハズレ値の影響を緩和できる
- 問題点: 通常、データは一つしかない
 - 対応策: Bootstrap 法により、データを複製する

3.5 シンプルな例

- データ = [5, 6, 100]
- 復元抽出により、同じ数(3)の事例をランダムに選ぶ
 - 複製データ 1 = [6, 6, 100] = の平均値 37.3
 - 複製データ 2 = [6, 6, 5] = の平均値 5.7
 - 複製データ 3 = [5, 5, 5] = の平均値 5

- ・最終予測 = 16
- ・ハズレ値(“100”)を反映しない予測も活用される

3.6 数値例



3.7 利点

- ・各複製データについて、ある事例が含まれる確率は 1/3 程度
 - ▶ 少数の事例に依存したモデルの比率は低い
 - より頑強なパターンの抽出が期待できる
- ・Random Forest: 回帰木を推定する際に、 X からランダムに選ばれた変数を除外する
 - ▶ 計算速度が向上し、推定精度も改善することが多い
- ・注: OLS などに対して、有効な方法ではない

4 モデル集計: 発展

4.1 予測モデルの選択

- ・OLS や Random Forest 等で推定した予測値のうち、どれを使用するのか?
 - ▶ 理論的に常に優れた方法は存在しない
- ・方法 1. 予測性能を評価し、最善のモデルを利用する
- ・方法 2. 予測値を集計

4.2 モデルの集計 (Stacking)

- 最終予測モデル

$$= \beta_{OLS} \times OLS\text{の予測}$$

$$+ \beta_{RF} \times RandomForest\text{の予測} + \dots$$

- β : 各予測結果を反映させる度合い
- 各予測値を”X”として用いた、線型モデル

4.3 推定方法

- データをサブデータ $\{1,..,G\}$ にランダム分割
- 第 1 サブデータ以外で予測モデルを推定し、第 1 サブデータを予測
- 第 2 サブデータ以外で予測モデルを複数推定し、第 2 サブデータを予測
- 以上を全てのデータについて繰り返す
- 予測対象 Y に対して、各予測値で回帰して β を推定

4.4 数値例: 3 分割

```
# A tibble: 9 × 3
  StationDistance     Price Group
            <int>    <dbl> <fct>
1             9   6.05     3
2             4   3.94     2
3             7  31.0      3
4             1   8.64     1
5             2  -5.99     3
6             7  -4.48     1
7             2  -0.895    1
8             3   0.00785   2
9             1  -3.12     2
```

4.5 数値例: Step 1

```
# A tibble: 9 × 5
  StationDistance     Price Group     OLS RandomForest
            <int>    <dbl> <fct>    <dbl>        <dbl>
1             9   6.05     3      NA        NA
2             4   3.94     2      NA        NA
3             7  31.0      3      NA        NA
4             1   8.64     1    -4.12      -1.89
5             2  -5.99     3      NA        NA
6             7  -4.48     1    12.9       16.7
7             2  -0.895    1   -1.29      -1.91
```

```

8          3  0.00785 2      NA      NA
9          1 -3.12    2      NA      NA

```

- Group 2,3 を Training データとして活用

4.6 数値例: Step 2

```

# A tibble: 9 × 5
  StationDistance     Price Group      OLS RandomForest
            <int>    <dbl> <fct>    <dbl>      <dbl>
1             9   6.05    3      NA      NA
2             4   3.94    2      4.86    -0.189
3             7  31.0     3      NA      NA
4             1   8.64    1     -4.12    -1.89
5             2  -5.99    3      NA      NA
6             7  -4.48    1     12.9     16.7
7             2  -0.895   1     -1.29    -1.91
8             3  0.00785  2      3.55    -0.189
9             1  -3.12    2     0.938     1.91

```

- Group 1,3 を Training データとして活用

4.7 数値例: Step 3

```

# A tibble: 9 × 5
  StationDistance     Price Group      OLS RandomForest
            <int>    <dbl> <fct>    <dbl>      <dbl>
1             9   6.05    3     -4.88    -1.84
2             4   3.94    2      4.86    -0.189
3             7  31.0     3     -3.03    -1.84
4             1   8.64    1     -4.12    -1.89
5             2  -5.99    3      1.61     0.945
6             7  -4.48    1     12.9     16.7
7             2  -0.895   1     -1.29    -1.91
8             3  0.00785  2      3.55    -0.189
9             1  -3.12    2     0.938     1.91

```

- Group 1,2 を Training データとして活用

4.8 数値例: Stacking

```
lm(Price ~ OLS + RandomForest, PopData)
```

```

Call:
lm(formula = Price ~ OLS + RandomForest, data = PopData)

```

Coefficients:		
(Intercept)	OLS	RandomForest
5.056	-1.248	0.243

- ω を非負、総和を 1 に基準化することも有効

4.9まとめ

- 伝統的な推定手法ではあまり用いられてこなかった、アイディアを用いた多くの手法が存在
- PC の処理能力の向上により、現実的な手法となる
- 常に上手くいく方法は現状存在しないので、複数の推定値の集計値を用いる方法を推奨
- 繙続学習用推奨資料: An Introduction to Statistical Learning

4.10 Reference

Bibliography