

スケートボーダー重心位置予測チャレンジ 3 位解法

2024年12月19日

DT-SN

アジェンダ

1. 自己紹介

2. 解法紹介

- ① データ補正
- ② 加速度学習
- ③ 平滑化
- ④ 球面座標系
- ⑤ データ分割
- ⑥ RegressorChain
- ⑦ 鏡像データ

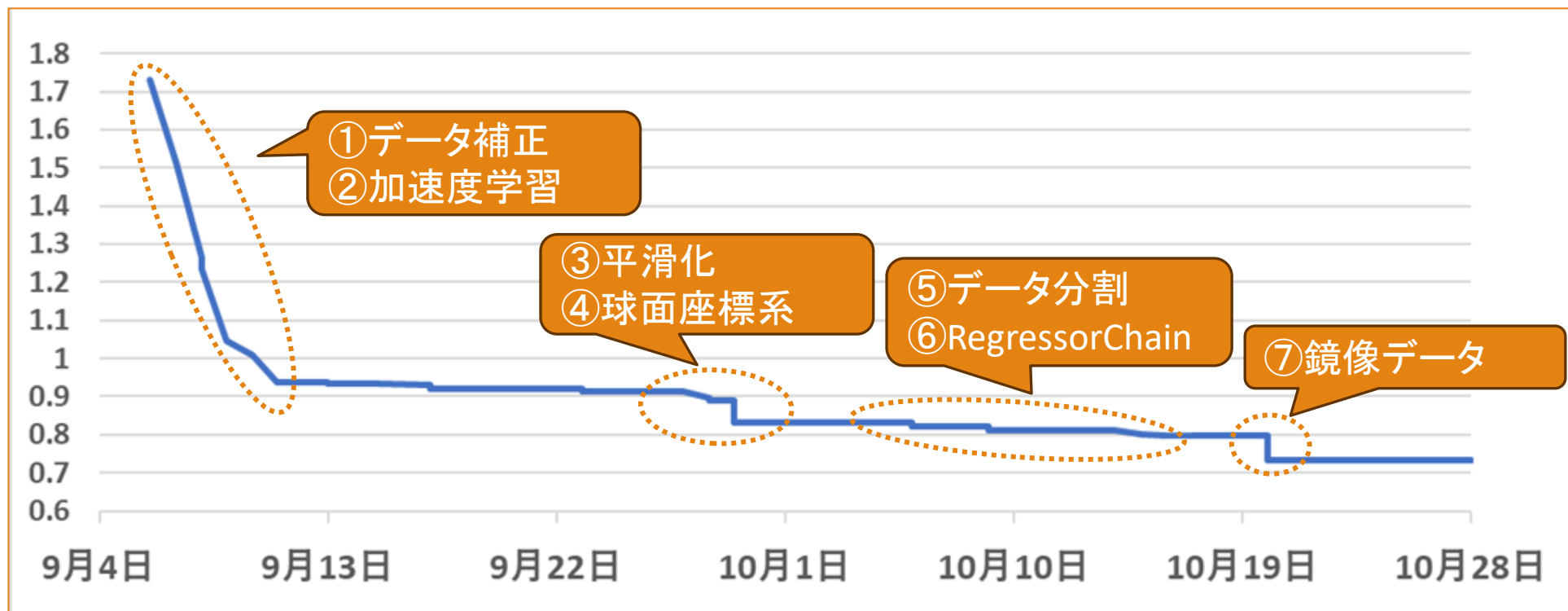
3. まとめ

1. 自己紹介

ユーザ名	DT-SN（中村 秀司）
所属	デンソーテクノ株式会社 セーフティソフトウェア事業部
業務履歴	<p>運転支援システム向け車両制御開発、シミュレータ開発</p> <p>↓</p> <p>運転支援システム向け車載カメラの画像認識アルゴリズム開発</p>

2. 解法紹介 スコア推移

■スコア推移（暫定評価0.7346682、最終評価0.7513308）

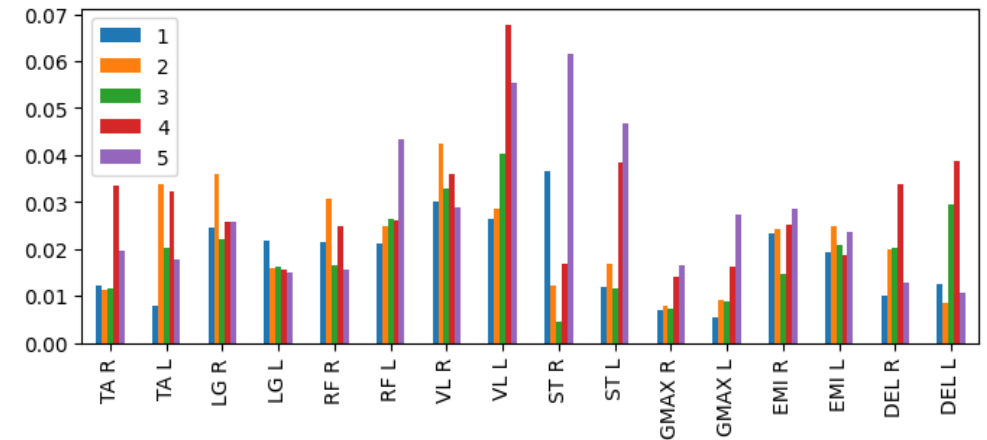


2. 解法紹介 ①データ補正

■筋電位の値域補正・周期変更

- 筋電位は正負がほぼ均等な波形（平均値はほぼ0）
→ **絶対値で扱う**
- プレイヤーごとに値域が異なる
→ **90パーセンタイルで割る**
- 目的変数と同じ周期(60Hz)のデータになるよう
33 or 34サンプル単位の平均と標準偏差に置き換え

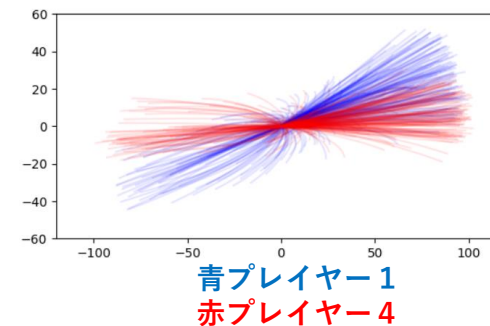
プレイヤーごとの平均値



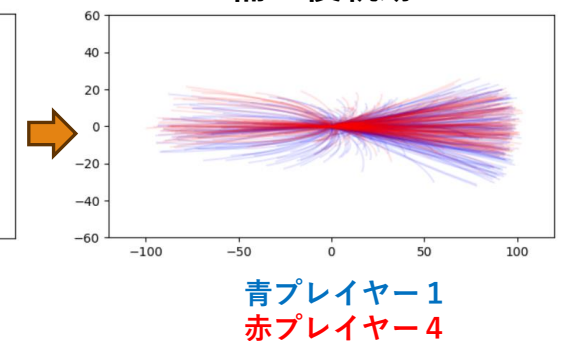
■重心速度の軸補正

- ランプ長辺方向がプレイヤーごとに異なる
→ 速度が一定以上 ⇒ 進行方向はランプ長辺方向と仮定し、**x軸と進行方向が平行になるように回転変換する**

補正前軌跡



補正後軌跡

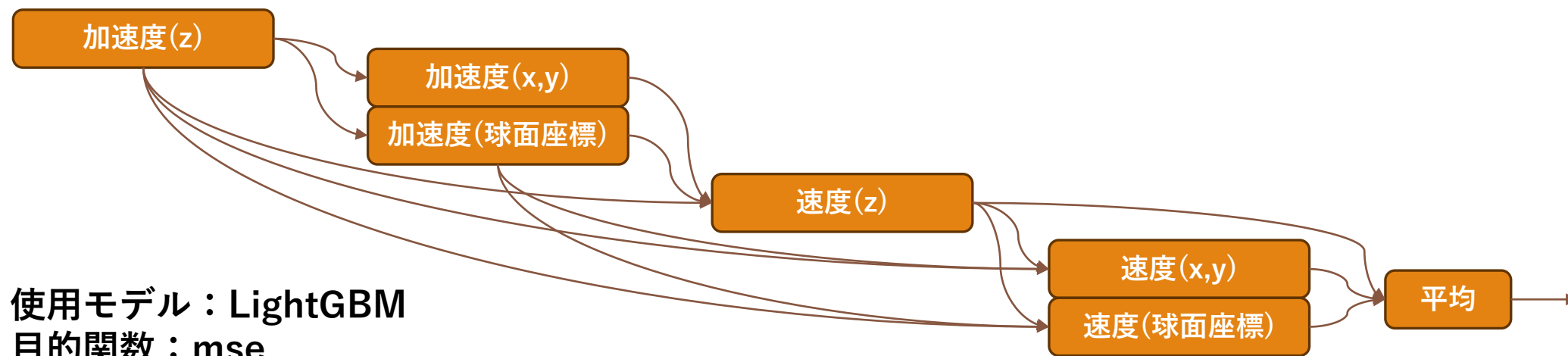


2. 解法紹介 ②加速度学習

■ 加速度変化から速度を予測する

- 筋電位から直接速度を予測するのは難しい
→ 加速度を予測し加速度の時系列変化から速度を予測する
- z 軸方向の動きは自分の意思でコントロールできるが x y 軸方向の動きには制約がある
→ z 軸方向を予測しその時系列変化から x y 軸方向を予測する

■ 学習/推論の流れ



2. 解法紹介 ③平滑化

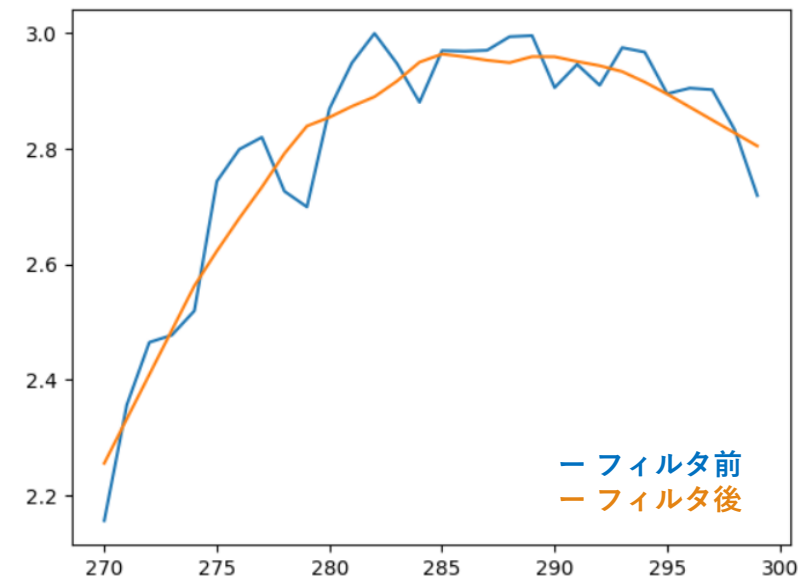
■推論結果のフィルタリング

コース形状からターン時以外は単調な加速度・速度変化をするはず

→ 推論結果をSavitzky-Golay（サビツキー・ゴーレイ）フィルタで平滑化する

■Savitzky-Golayフィルタの特徴

- ・データの点数と近似する多項式の次数だけで係数が決まる
- ・ピークが鈍りにくい
- ・処理が高速

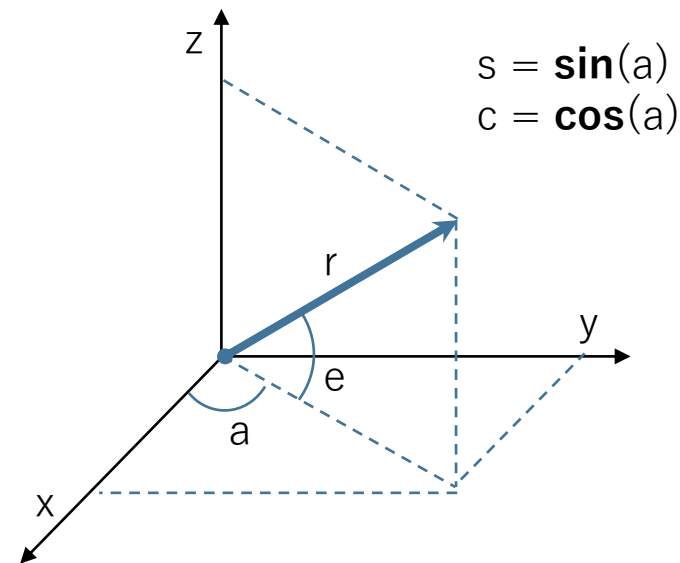
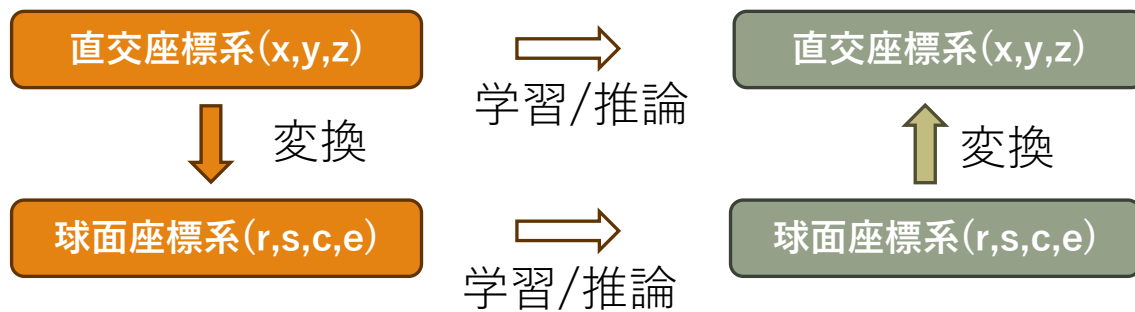


2. 解法紹介

④球面座標系

■球面座標系での学習/推論

スケートボードの操作は直進方向の加減速と方向回転 → 球面座標系と相性が良い
方位角は -180 度から 180 度の間で不連続になってしまうので \sin と \cos に置き換え
球面座標系で学習/推論した結果は直交座標系に変換し直交座標系の結果と平均を取る

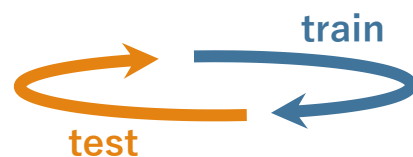


2. 解法紹介 ⑤データ分割

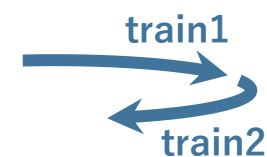
■trainデータ分割によるCV

trainのスロープ上りと下りを異なるfoldに分けることでtrainとtestの関係に近い状態のCVを行う
具体的には下図のように分割する

train1 (vel_x > 10)				train2 (vel_x <= 10)				ref.train	ref.test
1	2	3	4	1	2	3	4	5	5
cross validation(5fold)				train					
train				cross validation(5fold)				train	
train								holdout	train
train									holdout



⇒
近い状況を作る

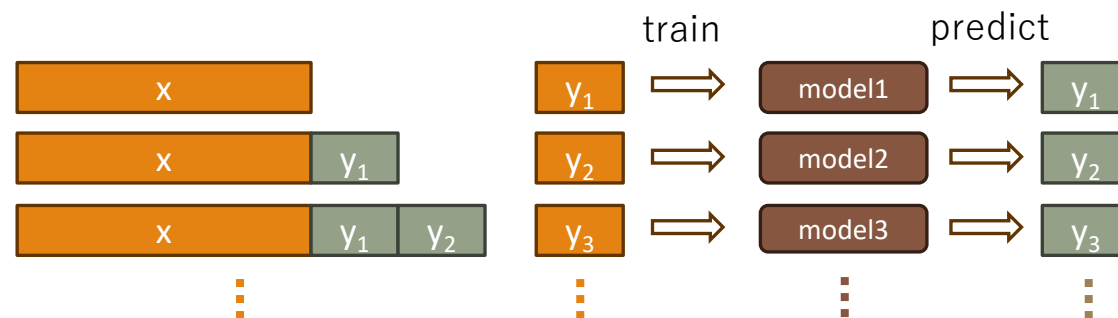


2. 解法紹介

⑥RegressorChain

■Regressor Chain

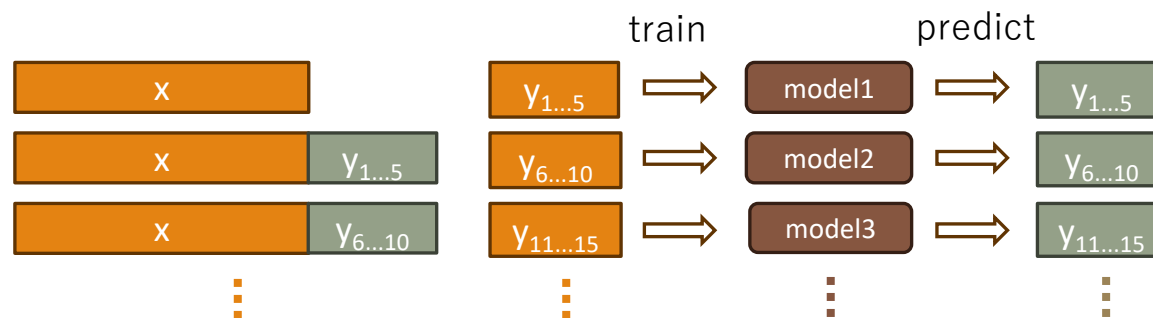
前周期の推論結果を特徴量に加え学習する
うまく使えば精度が上がるが
オーバーフィットしやすい



■オーバーフィット対策

対策1：複数周期まとめて学習/推論する

対策2：1つ前の周期のみ追加する



2. 解法紹介 ⑦鏡像データ

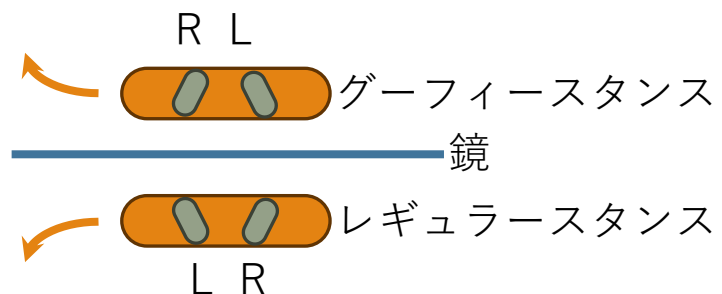
■鏡像データ追加

ターンが**パンピングの場合** → その後の滑走は**逆スタンスの状態**となる

train : スロープ下りで逆スタンス → train/testで差異が生じる
test : スロープ上りで逆スタンス

対策：鏡像データ(EMG左右反転とy軸速度反転したもの)を追加した

→ 鏡像データはスタンスを逆にしたものと解釈できるためtrain/testを差異を緩和できる



相違点

- ・左右の足が反転
- ・y軸の方向が反転

3. まとめ

■まとめ

下記がスコアアップに繋がった

- ・データをよく見て適切な補正する（値域、周期、座標軸、平滑化）
- ・予測しやすいところから順に攻める（加速度を予測し予測した加速度から速度を予測）
- ・train/testのデータの差異への対応（trainデータ分割によるCV、鏡像データ追加）

■反省点

- ・LightGBM以外のモデル（NN系など）を試すことができなかった
- ・筋電位を2000Hzのまま扱う良い手法は思いつかなかった

最後に、本コンペの関係者の皆様にお礼申し上げます