



# オンライン試験におけるカンニング検出システムの提案

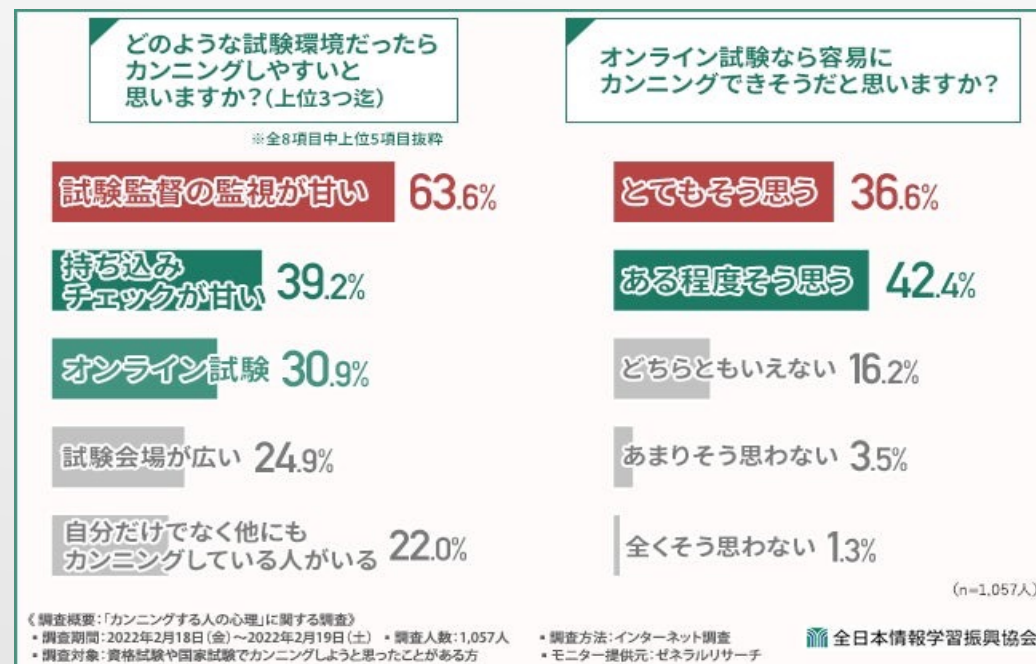
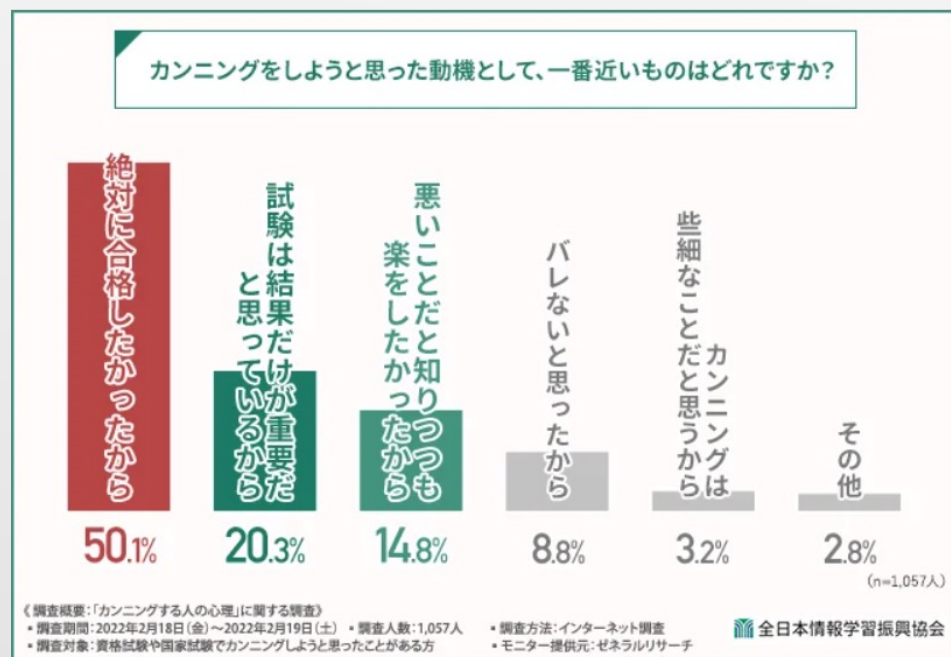
9BJM1211 倉持純也



# 研究背景

昨今のコロナ禍において感染拡大のためにオンライン試験が多くなっていると感じる。特にIT系の資格試験や留学に用いる語学試験、就職活動における適正検査等がオンラインでの受験が可能になっている。

これらの試験の点数や合否は受験者の今後の生活に大きく影響され、オンライン試験におけるカンニングを防ぐことは重要になってくる。



# 研究目的

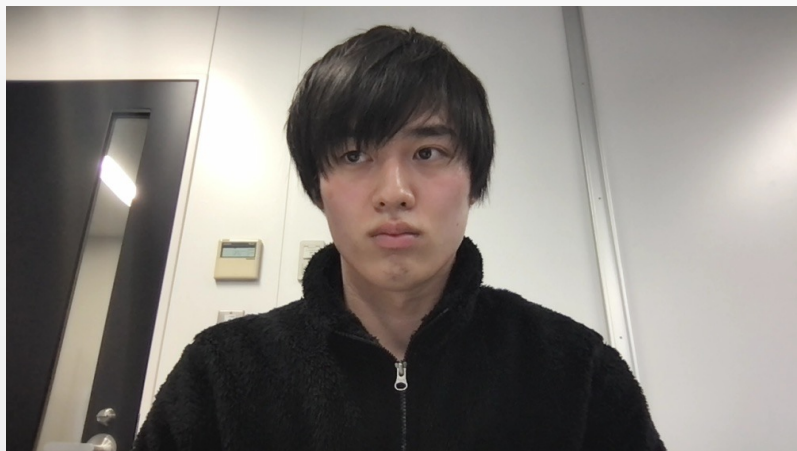
オンライン試験は試験者にとってはカンニングがしやすいそうだという認識だったが、本研究でオンライン試験におけるカンニング検出を可能だという事を明らかにする。

具体的には機械学習モデルを用いることで試験者の行動パターンからパソコンの画面に視線を送っているのかを検出する。

従来研究（顔動画像データを用いたオンライン試験におけるカンニング検出）では、複数の機械学習モデルを作成して精度の比較をしていたが、本研究では実際に動作した際のフレームレートの比較まで行い、リアルタイムでカンニングの検出を可能であることを示す。そこで本研究ではオンライン試験における受験者の顔表情が映されている映像を用いてリアルタイムにカンニングを検出することが可能なシステムを提案することを目的とする。

# 研究仮説

カンニングとして疑わしい行為



目線を外している



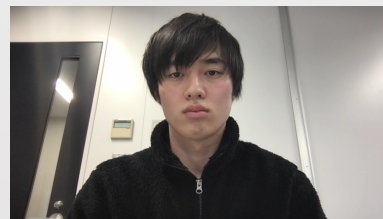
顔を傾けている



会話をしている

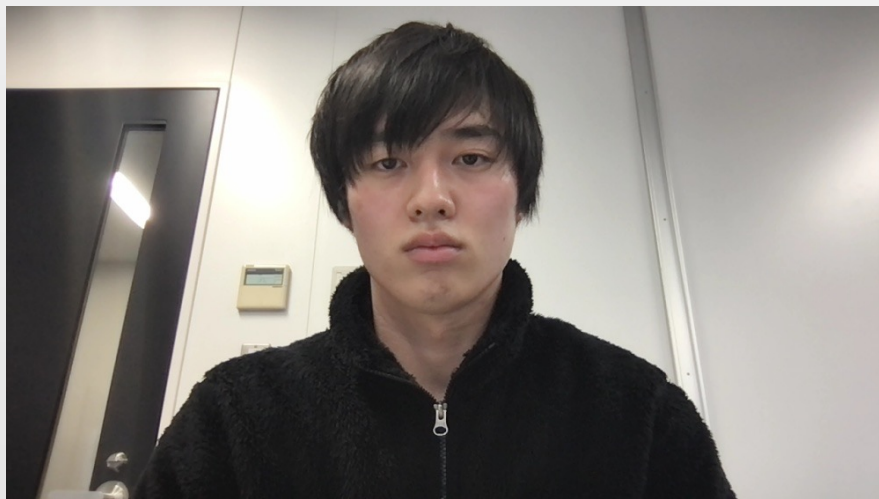


カンニングしている？していない？

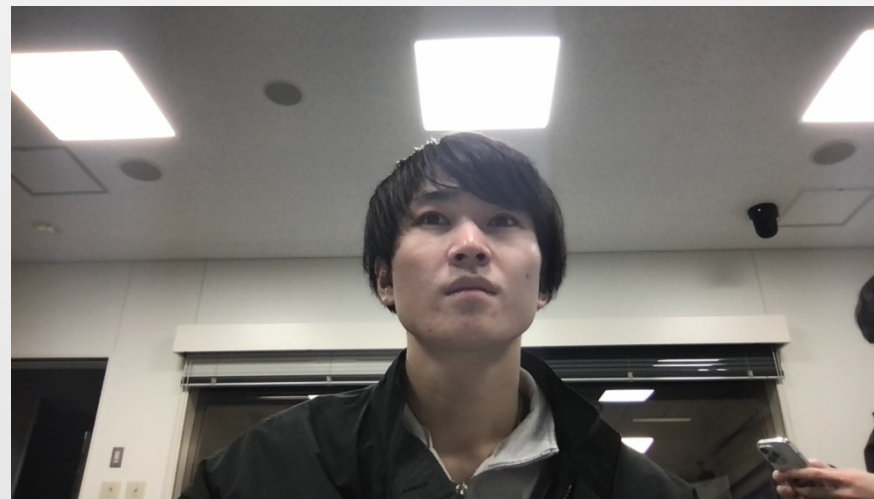


# 研究方法

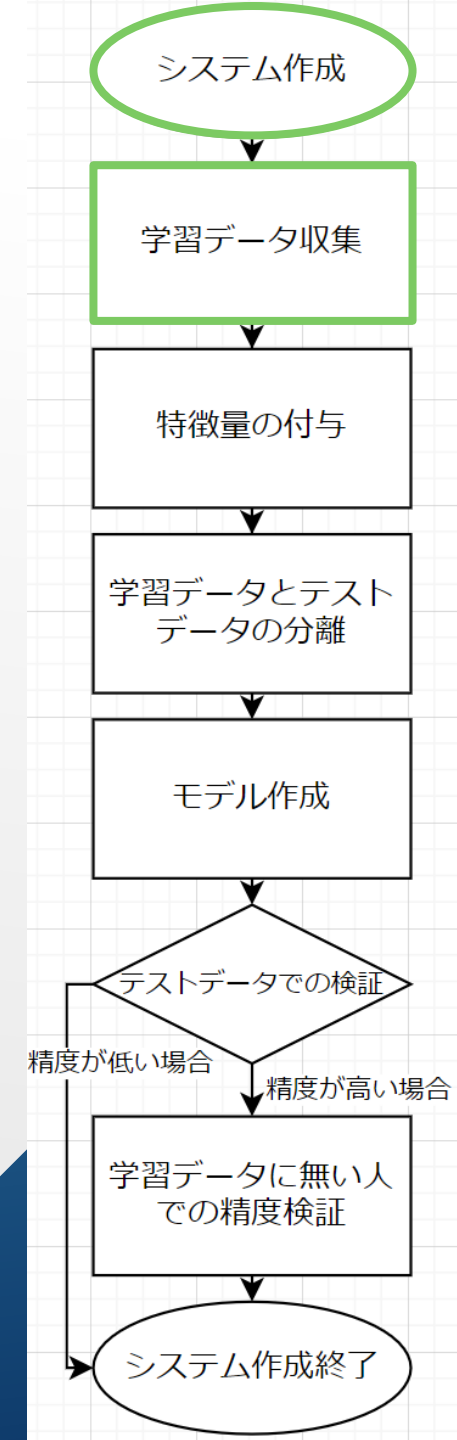
学習データ収集ではオンライン試験を今まで受験したことがある学生7人のPCの画面を見ている画像とみていない画像を約100枚ずつ撮影を行い、合計で約1400枚の画像を使用した。



画面を見ている



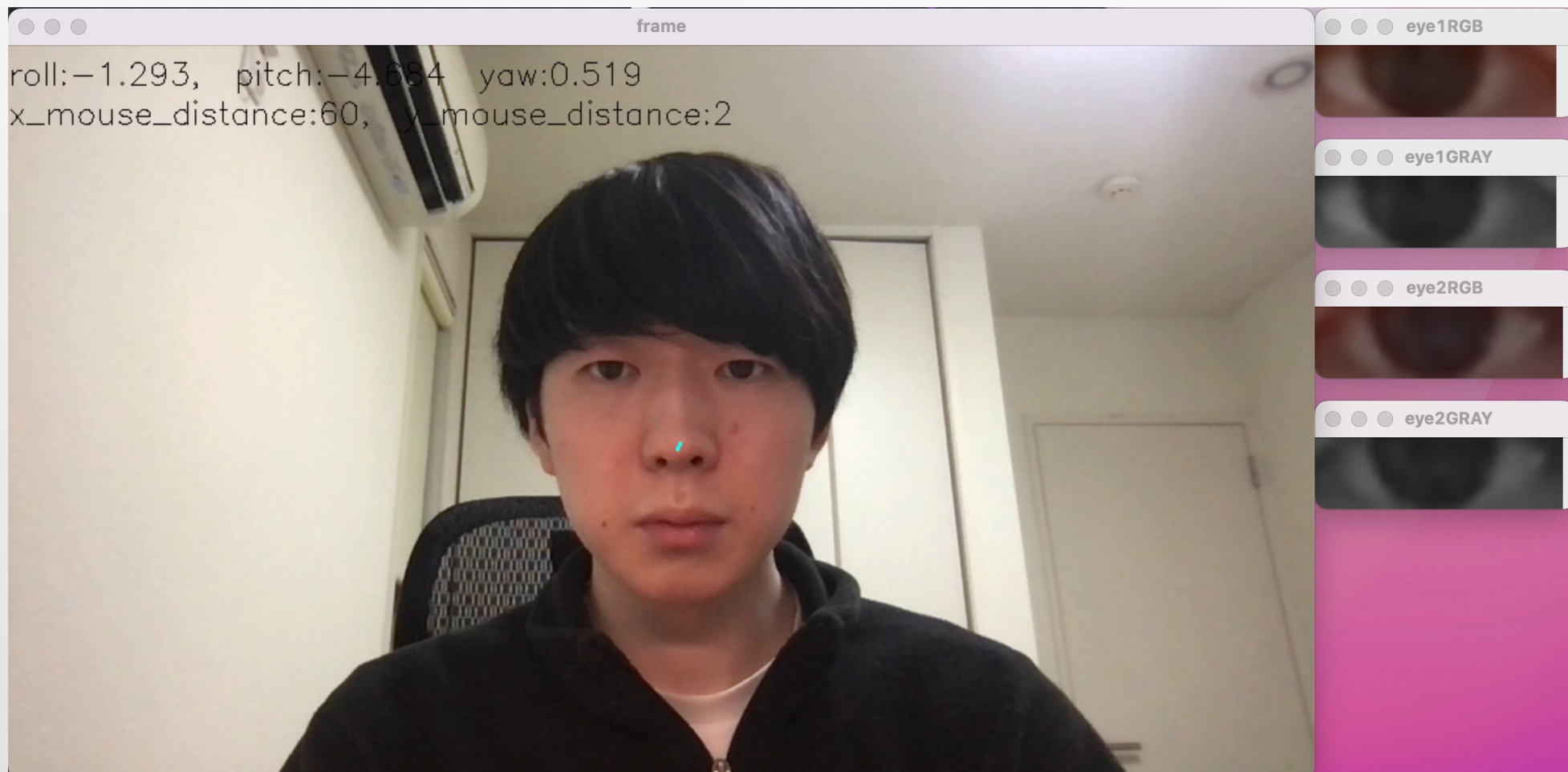
画面を見ていない



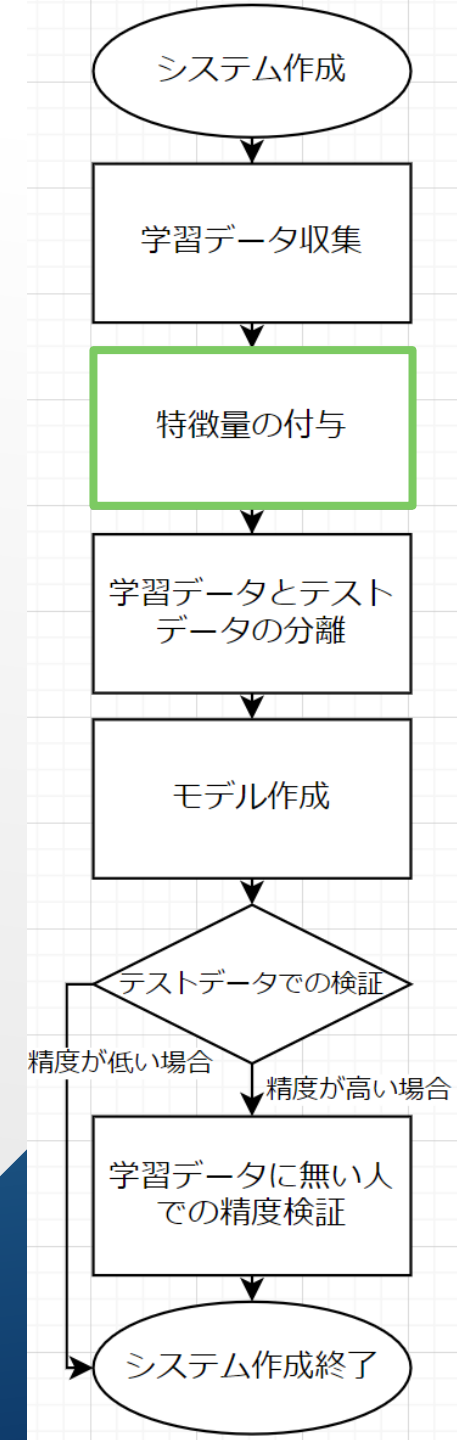


Dlibを用いて顔にランドマークとして座標を配置する。得られた座標より

- ・ 顔の傾きであるロール、ピッチ、ヨーを求める
- ・ 両目の白黒画像化を行う
- ・ 口元の開き具合の計算処理を行う



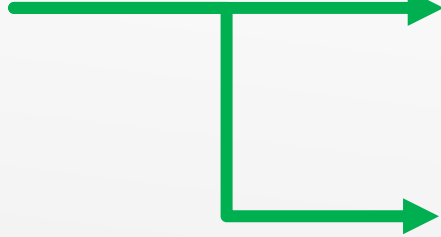
例：リアルタイムでの特徴量の付与の可視化



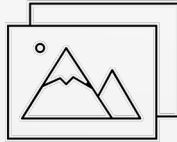
## 集めたデータの分離



約1400枚  
の画像



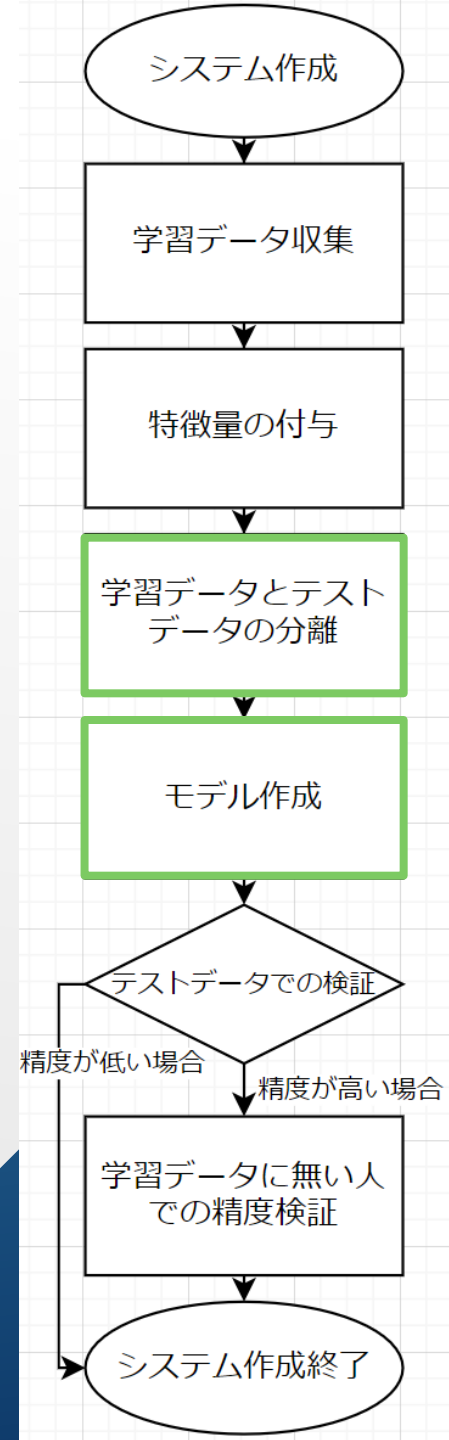
学習用データ：約1050枚の画像



テスト用データ：約350枚の画像

K-分割交差検証を用いて学習用データからさらに学習用と検証用に分けたのち、カンニング判定を行う以下のモデルを作成

- ロジスティック回帰
- ナイーブベイズ
- 決定木
- ランダムフォレスト
- k-最近傍法
- SVM linear
- SVM rbf
- LightGBM
- TensorFlow/TensorFlow(Optuna使用)

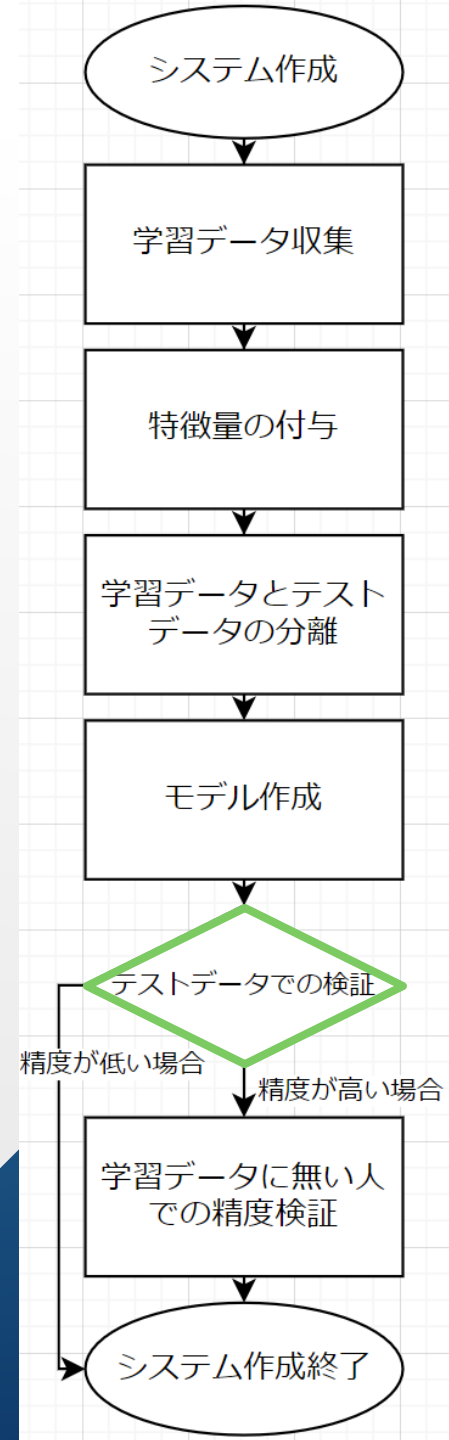


|                     | 正答率   | 適合率   | 再現率   | F値    |
|---------------------|-------|-------|-------|-------|
| ロジスティック回帰           | 0.834 | 0.842 | 0.811 | 0.827 |
| ナイーブベイズ             | 0.749 | 0.806 | 0.639 | 0.712 |
| 決定木                 | 0.887 | 0.898 | 0.868 | 0.882 |
| ランダムフォレスト           | 0.958 | 0.956 | 0.960 | 0.958 |
| k-最近傍法              | 0.906 | 0.976 | 0.827 | 0.896 |
| SVM linear          | 0.840 | 0.860 | 0.802 | 0.830 |
| SVM rbf             | 0.927 | 0.950 | 0.896 | 0.922 |
| LightGBM (Optuna)   | 0.868 | 0.898 | 0.822 | 0.858 |
| TensorFlow          | 0.936 | 0.936 | 0.933 | 0.934 |
| TensorFlow (Optuna) | 0.963 | 0.967 | 0.957 | 0.961 |

正答率→画面の注視状況をどちらとも正確に判断できている

適合率→値が低いほど画面を注視していると誤りがち

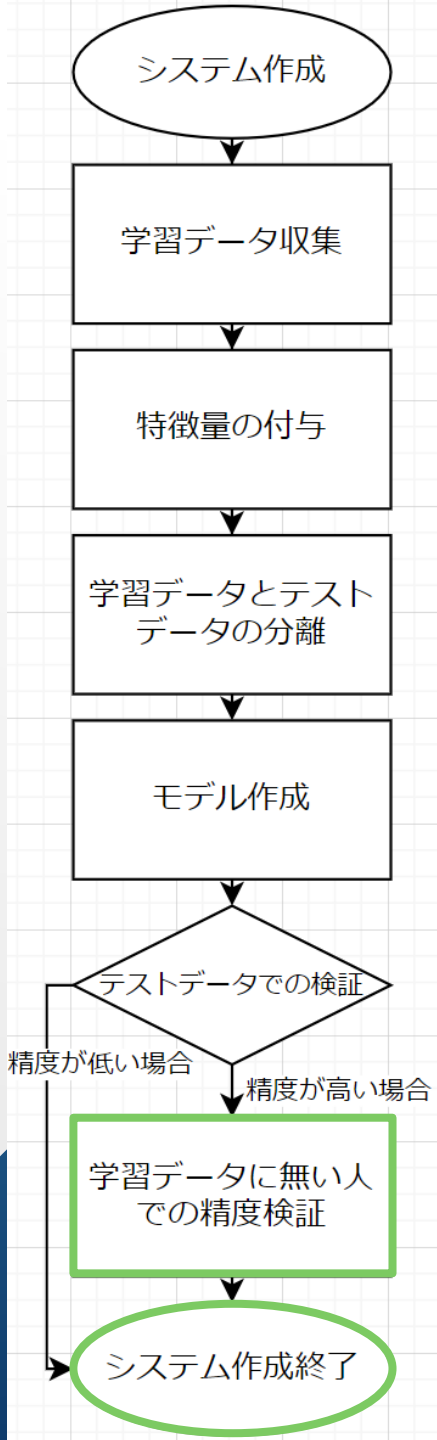
再現率→値が低いほどカンニングしていると誤りがち





# 学習データに無い新しい被験者2人を用いて検証を行った

|                     | 正答率         | 適合率         | 再現率       | F値          |
|---------------------|-------------|-------------|-----------|-------------|
| ランダムフォレスト           | 1人目 : 0.555 | 1人目 : 0.529 | 1人目 : 1.0 | 1人目 : 0.692 |
|                     | 2人目 : 0.930 | 2人目 : 0.877 | 2人目 : 1.0 | 2人目 : 0.935 |
|                     | 平均 : 0.743  | 平均 : 0.660  | 平均 : 1.0  | 平均 : 0.795  |
| SVM rbf             | 1人目 : 0.720 | 1人目 : 0.641 | 1人目 : 1.0 | 1人目 : 0.781 |
|                     | 2人目 : 0.995 | 2人目 : 0.990 | 2人目 : 1.0 | 2人目 : 0.995 |
|                     | 平均 : 0.858  | 平均 : 0.778  | 平均 : 1.0  | 平均 : 0.875  |
| TensorFlow (Optuna) | 1人目 : 0.610 | 1人目 : 0.562 | 1人目 : 1.0 | 1人目 : 0.719 |
|                     | 2人目 : 1.0   | 2人目 : 1.0   | 2人目 : 1.0 | 2人目 : 1.0   |
|                     | 平均 : 0.805  | 平均 : 0.719  | 平均 : 1.0  | 平均 : 0.837  |



## 考察

TensorFlowでは1人目と2人目の差が大きく、過学習の影響を受けている。しかし、すべてのモデルにおいて再現率が1.0であったことからカンニングをしていなくても誤ってカンニングとして判定する恐れはあるが、逆にカンニング行為は1度も見逃さないことになる。そのため、カンニング判定のシステムとしては有効であると考えられる。

# 結論

