

キノコの毒性の有無を
即座に見分けよう！

2-3-27 高瀬 歩



1. 毒を見分ける方法
2. 結果
3. 結論
4. 感想



1. 毒を見分ける方法

有名な毒キノコたち

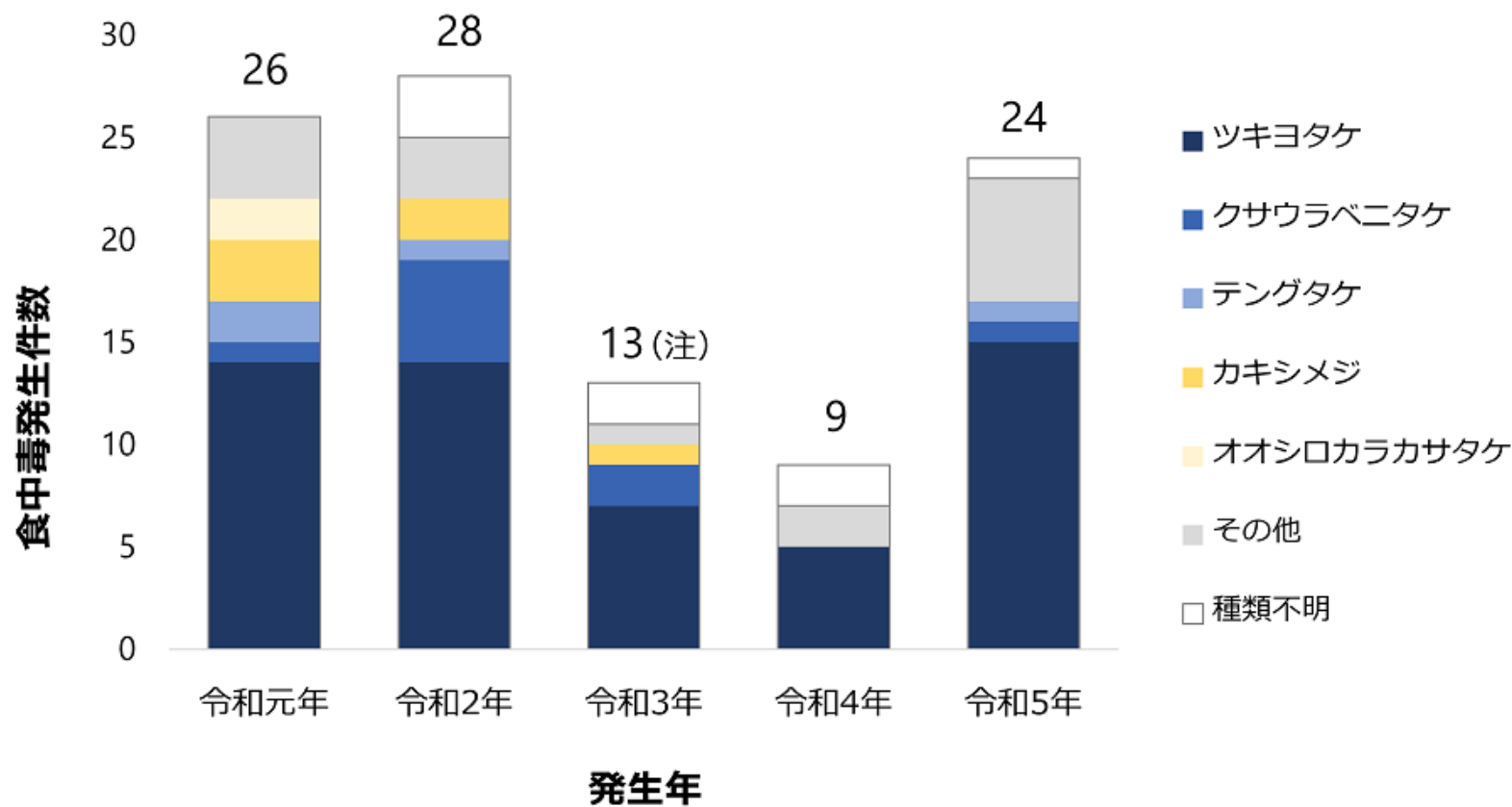
- ・ベニテングダケ



- ・ドクツルタケ



毒キノコによる事故数



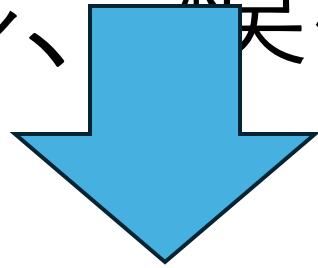
注：令和3年の食中毒発生事例は12件であるが、うち1件はクサウラベニタケ及びイッポンシメジの両方が原因と推定されており、本グラフでは毒きのこの種類ごとに発生件数1件としてカウントした。



毒キノコ、一目で見分けられた
らとってもすごいのでは？



毒キノコならではの特征があるはず、、
(色、形、模様など)



その特徴を**決定木**で分析する



- 今回用いるライブラリは授業でも扱ったrpart

- データはUCIのデータを用いる

<https://archive.ics.uci.edu/dataset/73/mushroom>



UCIのデータについて

- キノコの特徴について22項目で区別されている
- 項目ごとに一文字に省略されており、少々読み取りが難しい
- 欠損値がある



Variables Table

Variable Name	Role	Type	Description
poisonous	Target	Categorical	
cap-shape	Feature	Categorical	bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s
cap-surface	Feature	Categorical	fibrous=f,grooves=g,scaly=y,smooth=s
cap-color	Feature	Binary	brown=n,buff=b,cinnamon=c,gray=g,green=r, pink=p,purple=u,red=e,white=w,yellow=y
bruises	Feature	Categorical	bruises=t,no=f
odor	Feature	Categorical	almond=a,anise=l,creosote=c,fishy=y,foul=f, musty=m,none=n,pungent=p,spicy=s



2. 結果

実際にrpartで分析を行っていく

```
# キノコのデータを格納しているURL
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/mushroom/agaricus-lepiota.data"

library(rpart)
library(rpart.plot)

# 列名を並べる
# 1番目が毒性の有無、2番目以降が特徴
column_names = c("class", "cap-shape", "cap-surface", "cap-color", "bruises",
                  "odor", "gill-attachment", "gill-spacing", "gill-size",
                  "gill-color", "stalk-shape", "stalk-root", "stalk-surface-above-ring",
                  "stalk-surface-below-ring", "stalk-color-above-ring",
                  "stalk-color-below-ring", "veil-type", "veil-color", "ring-number",
                  "ring-type", "spore-print-color", "population", "habitat")

# データを読み込む
mushroom_data = read.csv(url, header = FALSE, col.names = column_names, na.strings = "?")
```



下記のように決定木をplotしてみる

```
# 欠損値を含むキノコ（行）を除外
```

```
mushroom_data_clean = na.omit(mushroom_data)
```

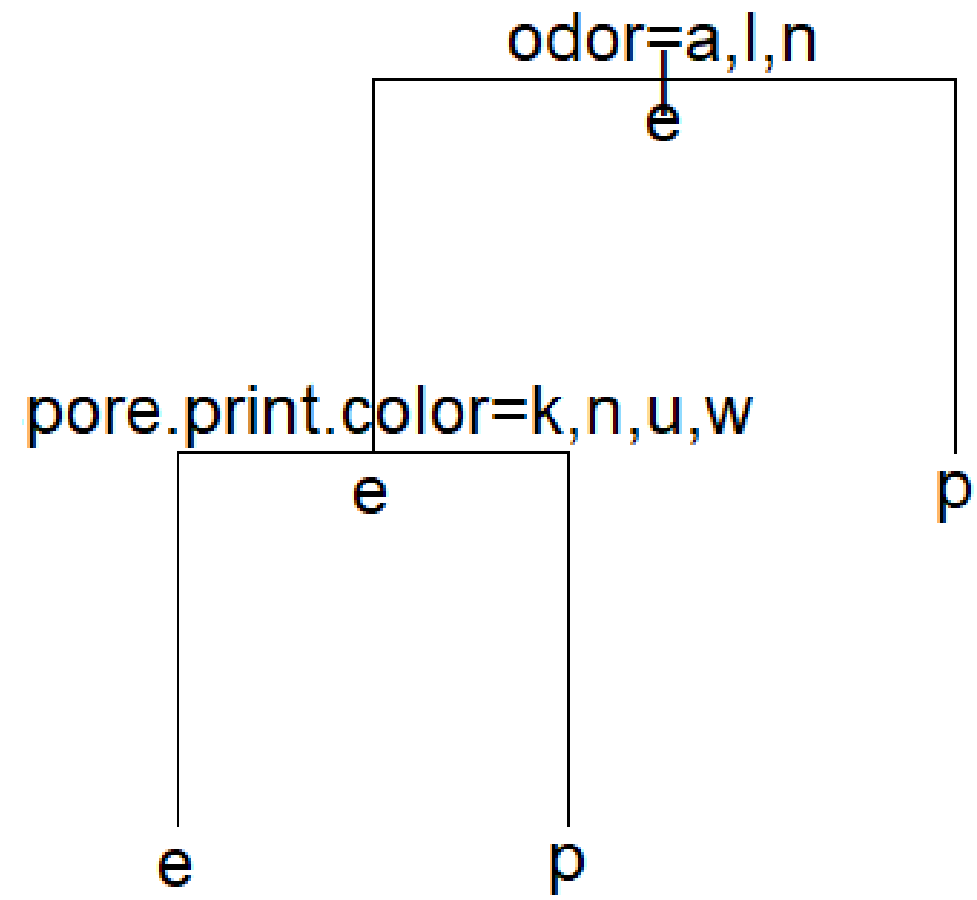
```
# 決定木モデルを作成する
```

```
model = rpart(class ~ ., data = mushroom_data_clean, method = "class")
```

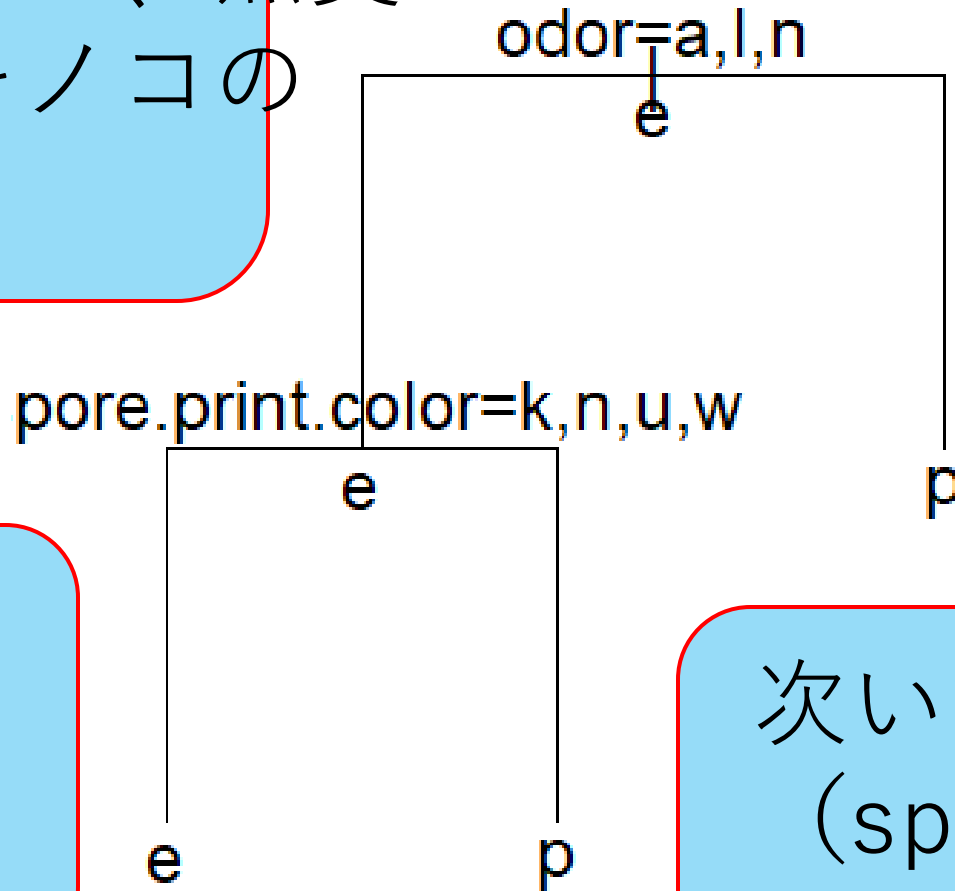
```
plot(model, uniform=T, branch = 1, margin = 0.15)
```

```
text(model, all = T, pretty = 0)
```





匂い (odor) が
アーモンド、アニス、無臭
以外の場合、毒キノコの
可能性が高い

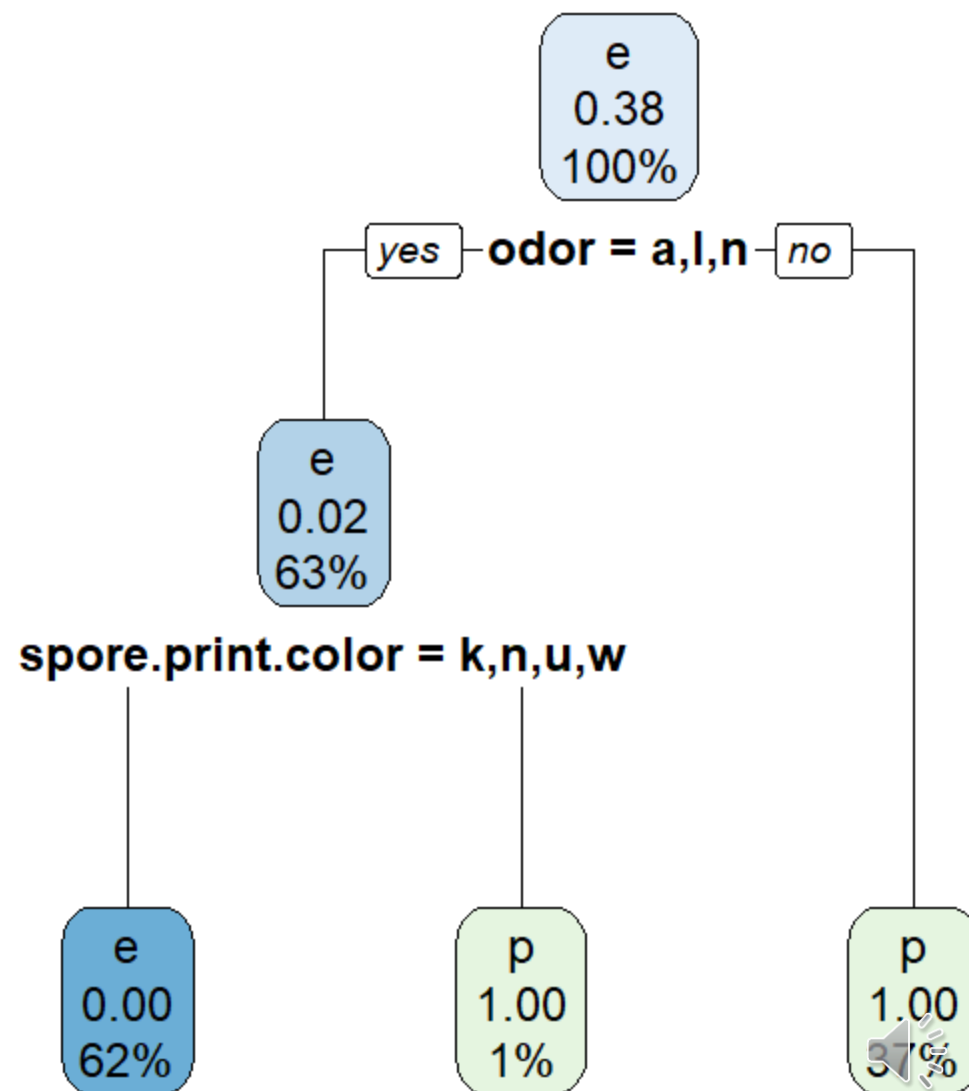


この決定木では
e=edible
p=poisonous

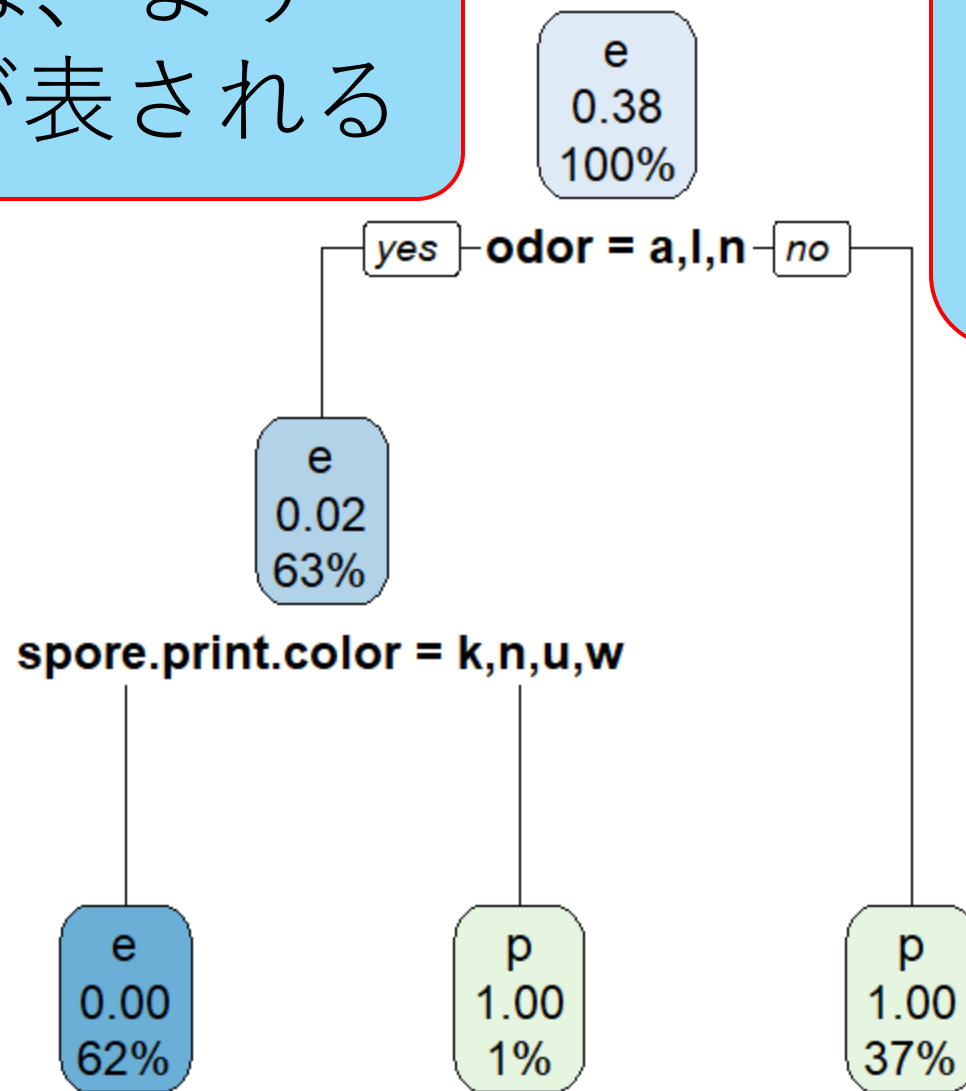
次いで、胞子の色
(spore-print-
color) が重要 

【おまけ】 Rpart.plot(
を用いてより見やすい
図を作ってみる

```
# 決定木を更に見やすく \  
rpart.plot(model)
```

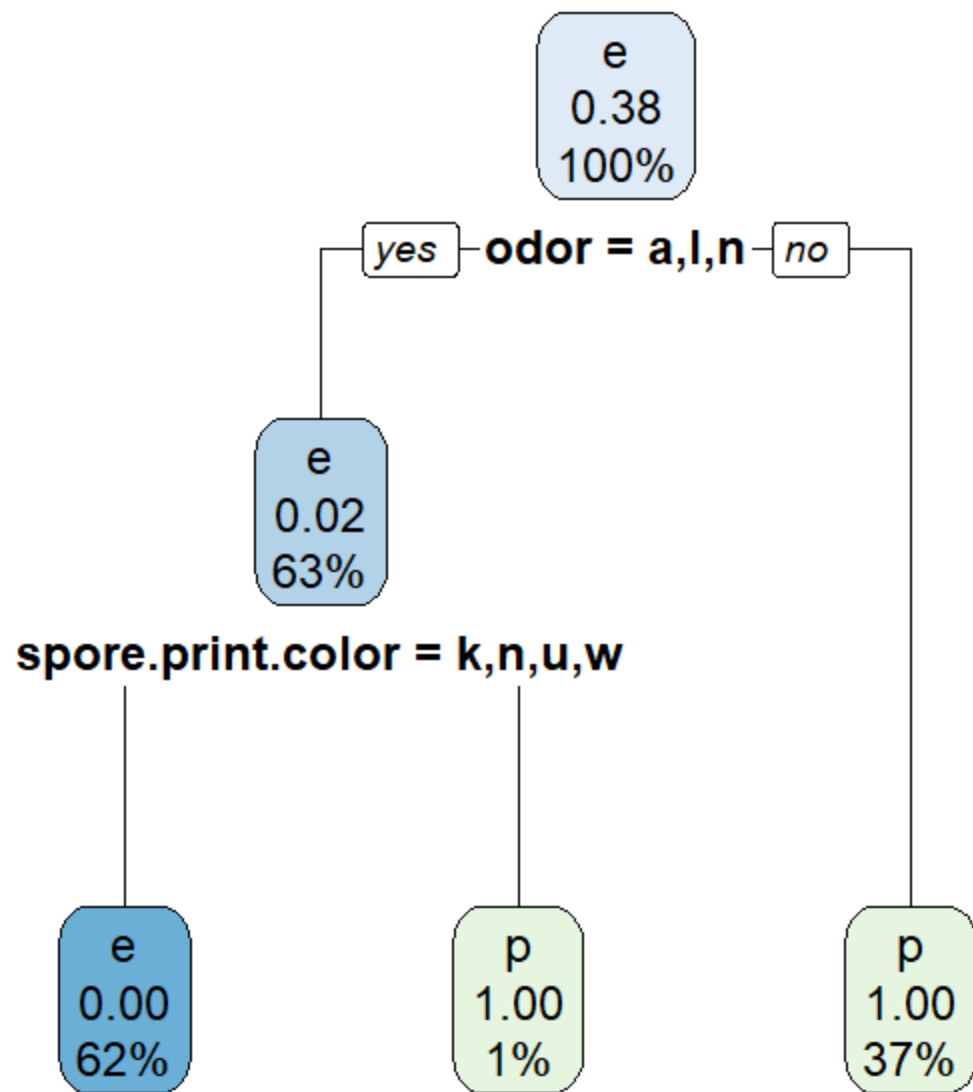


枠内のeやpは、より
多数派の方が表される



真ん中の数字は、
毒キノコが含まれ
る確率を表してい
る

最後に1番下の数
字は、そのノード
に該当するデー
タの割合を表す



【この決定木から分かることの
最初の何の条件もない状態
では毒の確率は38%だが、
匂いの条件で分けるだけでも
2%まで激減する



2. 結論

- 先ほどの決定木から分かるように、毒キノコの判別大きく影響しているのは匂いであり、次に胞子の色が影響している
- また、このデータがどれほど正しいか、適合度を求めることで判別する



判別したデータの適合度を調べる

混同行列を作成

```
confusion_matrix = table(Actual = mushroom_data_clean$class, Predicted = predictions)  
print(confusion_matrix)
```

混同行列→元データと決定木で得られた結果（予測）の間で、どれほどの相違があったか一目で見られる

```
> print(confusion_matrix)
```

	Predicted	
Actual	e	p
e	3488	0
p	16	2140



混同行列から適合度を計算する

適合度→作成した統計モデルが、元データと比べてどれほど正しいかを表す数値

```
> # モデルの適合度を計算
```

```
> accuracy = sum(diag(confusion_matrix)) / sum(confusion_matrix)
```

```
> print(paste("モデルの適合度:", accuracy))
```

```
[1] "モデルの適合度: 0.997165131112686"
```

これにより、今回作成したモデルはかなり正確であるといえる



4. 感想

- 授業内で学んだライブラリの使い方を復習し、何が最も適しているかを吟味したことから、何を用いるべきかを判断するのに今後役立つと感じた
- 始めて自分で興味のあるデータを用いて学習したが、その楽しさをはっきりと実感できた

