言語処理課題2019

岡山大学　竹内孔一

レポート

- 著者推定問題をSVMで解いて実行結果を報告
- 自分で工夫した点があれば記述
- SVM (pythonのモジュール) に対する入力形が何か説明する

データは演習室のマシン

/home/users/edu2019/lect/tk/nlsample/novel

SVMの実行は演習室でも google colaboratoryでもかまわない

以降のスライドと演習室においてあるデータを見て
レポートを作成して下さい

締めきりは 7月 24日 (水)

文書分類を試してみる

■ 文書分類　　青空文庫 & NPCMJ

文の著者を推定してみよう!

0　a) 芥川龍之介　アグニの神
1　e) 江戸川乱歩　押絵と旅する男
2　m) 森鴎外　　鼠坂

a,婆さんはどこからとり出したか、眼をつぶった妙子の顔の先へ、一挺のナイフを突きつけました。
e,私は仕方がないので母親に貰ったお小遣いをふんぱつして、人力車に乗りました。
m,小川君は好奇心が起って溜まらなくなった。
a,一そ警察へ訴えようか？
a,イツモダト私ハ知ラズ知ラズ、気ガ遠クナッテシマウノデスガ、今夜ハソウナラナイ内ニ、ワザト魔法ニカカッタ真似ヲシマス。

形式　正解タグ，小説の1文

文書分類

■ 機械学習 (深層学習を含む)

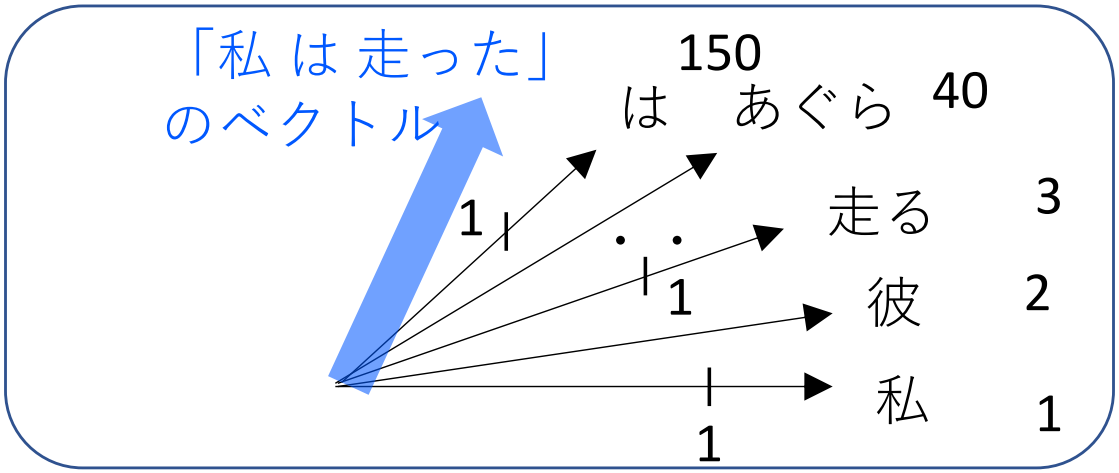
(1) 入力文を単語毎に1座標としてベクトル化　(Bag-of-words)

文書

全単語に番号

座標軸の番号

1: 私
2: 彼
3: 走る
...
N: <unk>



1文をN次元ベクトルで表現

「私は走った」➔「私 / は / 走る」➔ { 1:1 3:1 150:1}

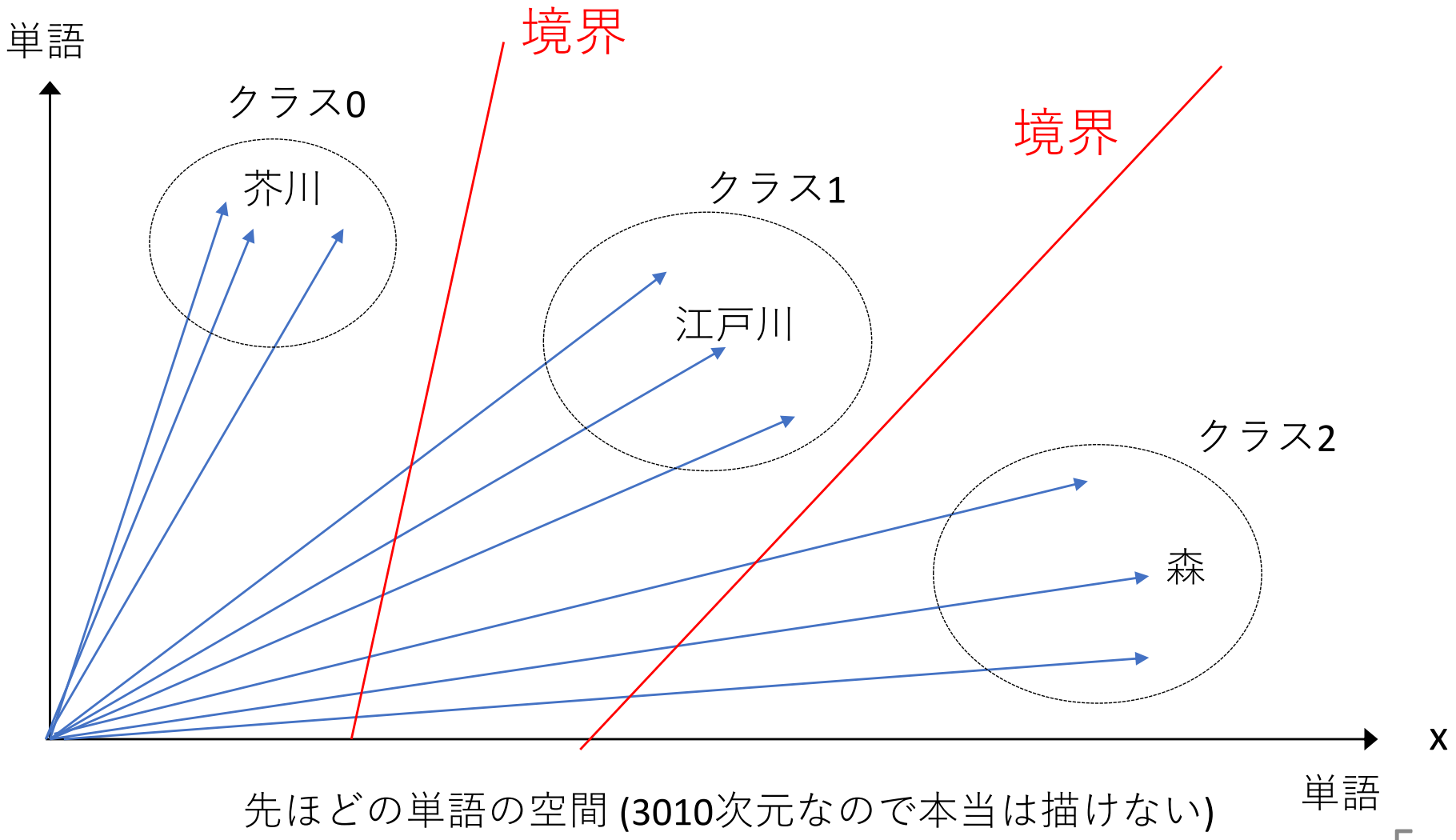
特徴: 1文は必ず固定長のベクトル(N次元)
ベクトル内は複数「1」が立つ (回数でもよい)
文内の単語の順序は無視

事例は全部で3010語

文書分類/機械学習/学習

■ 学習とは?

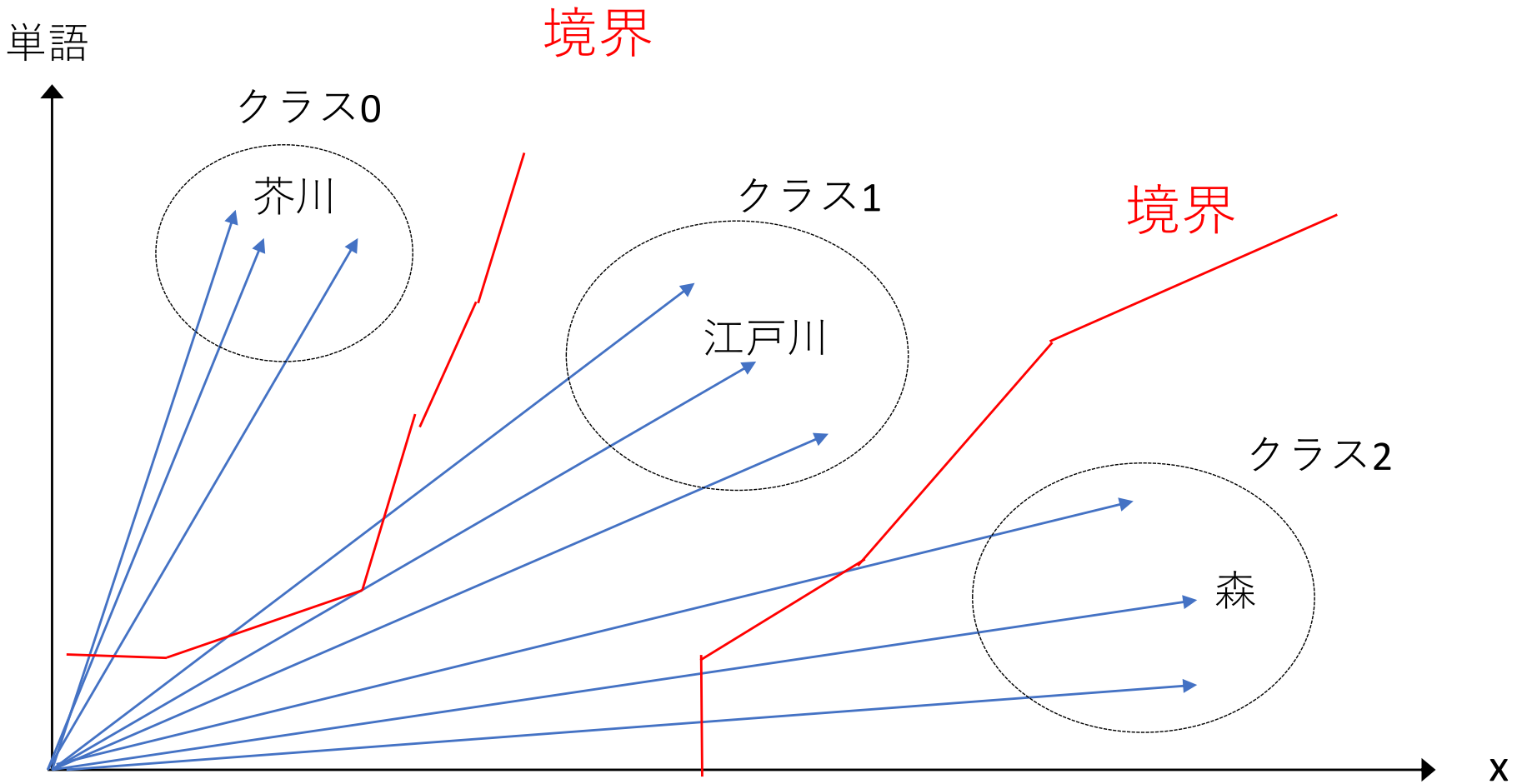
学習データをうまく分ける境界を見つけること!



先ほどの単語の空間 (3010次元なので本当は描けない)

文書分類/機械学習/学習

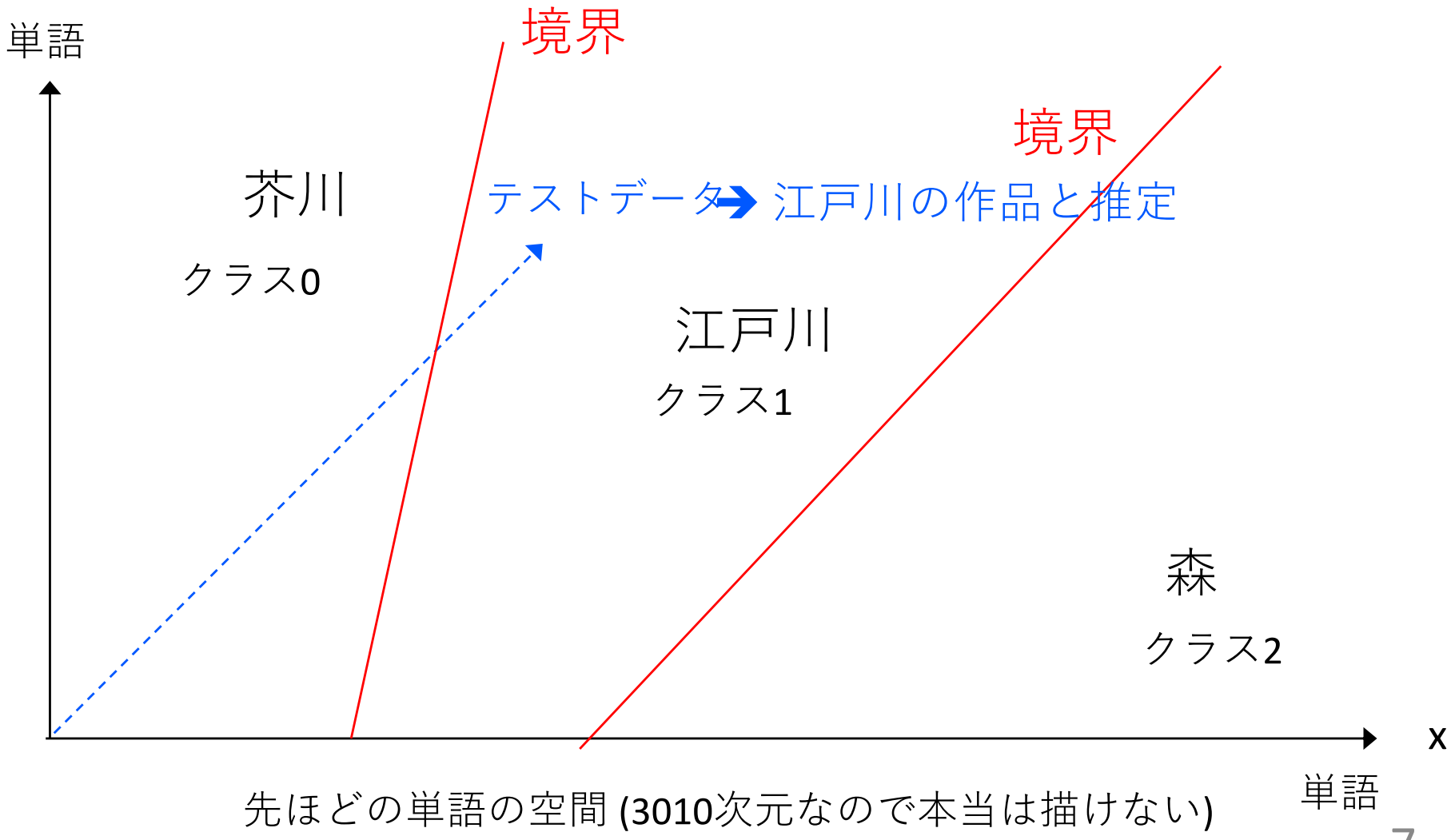
■ 深層学習も同じ　ただし，境界がどんな形でも作れる!



先ほどの単語の空間 (3010次元なので本当は描けない)

文書分類/機械学習/推定

■ 推定　テストデータを境界に合わせて分類



文書分類/実習

■ 学習データとテストデータ

学習データ とテストデータに分ける

学習データでモデルを作成

評価はテストデータで行う

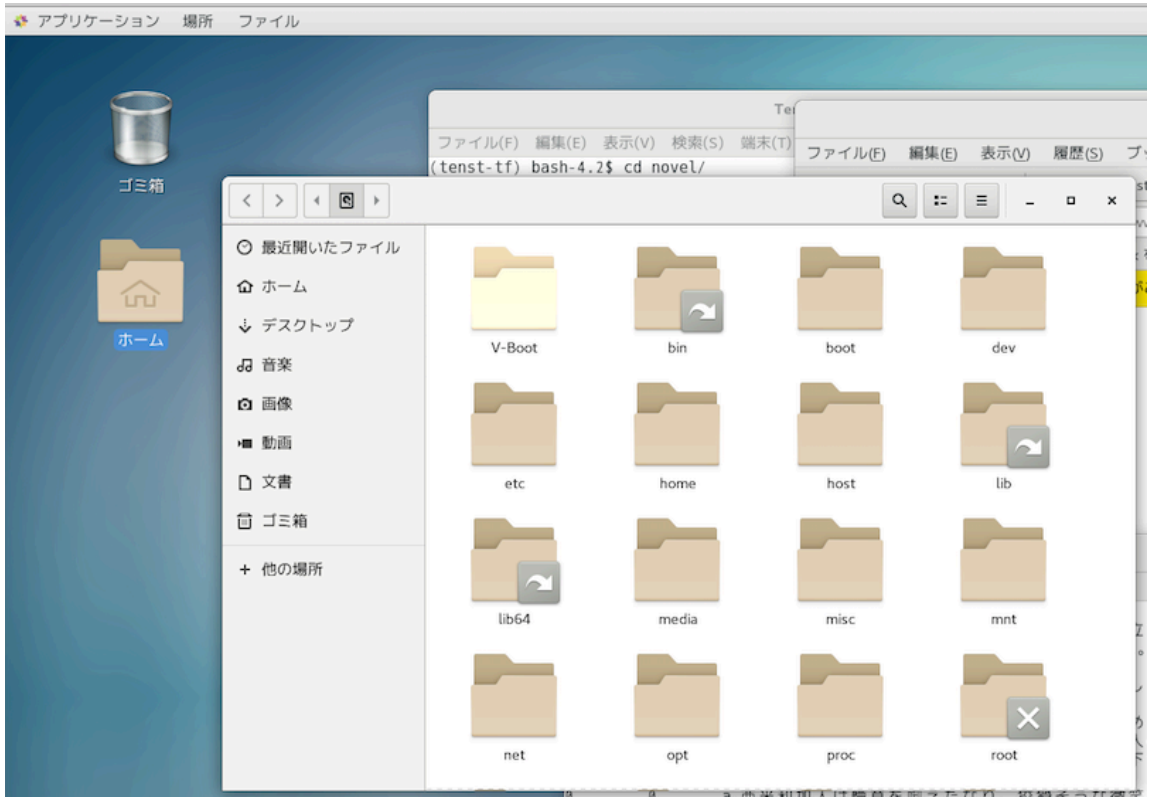
文書分類/実習

■ データの確認

データ /home/users/edu2019/lect/tk/nlsample/novel/data

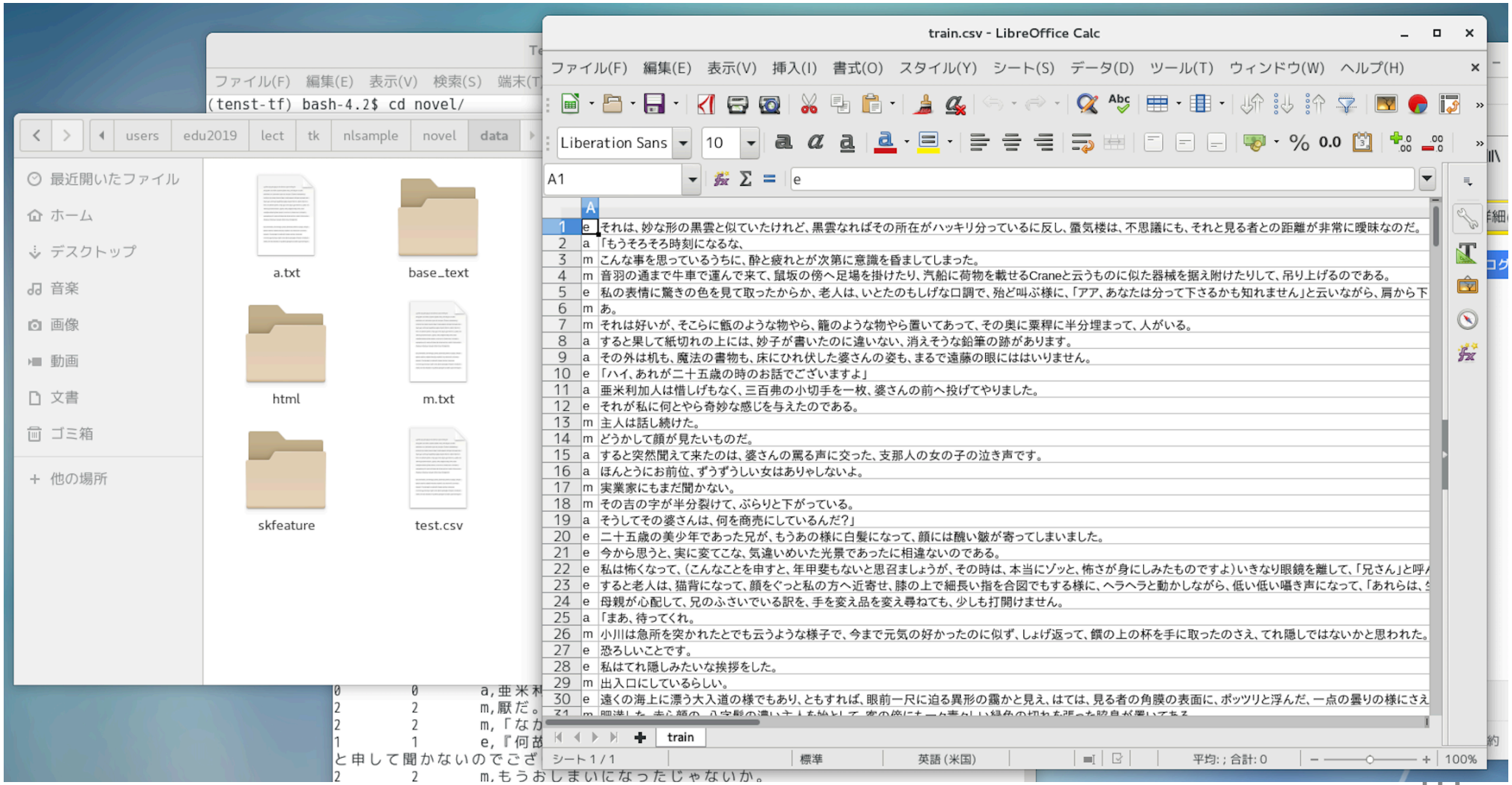
学習データ train.csv　　テストデータ test.csv

データを見てみよう!!



文書分類/実習

train.csv または test.csvをダブルクリックすると シートを確認でき



文書分類/実習

■ ベクトル化したデータの確認 ←これが本当の入力

/home/users/edu2019/lect/tk/nlsample/novel/data/feature

学習用，テスト用のベクトル化データ　train.feature と test.feature

■特徴抽出の中身を見る (見たい方はどうぞ)

```
$ cd  /home/users/edu2019/lect/tk/nlsample/novel/data/feature
$ lv  train.feature
```

分類

```
1 649:1 1337:1 806:1 2578:1 1599:1 1615:1 1553:1 2593:1 283:1 1708:1 2211:1 1280:1
2608:1 844:1 806:1 2593:1 717:1 343:1 817:1 1844:1 2888:1 1388:1 1864:1 2211:1 1280:1
360:1 2262:1 806:1 2757:1 1337:1 806:1 1881:1 360:1 1934:1 806:1 649:1 283:1 2015:1
2807:
1 283:1 1553:1 1555:1 2888:1 1928:1 360:1 0:1 1599:1 1553:1 1599:1 890:1
0 1376:1 254:1 1283:1 303:1 360:1 717:1 2794:1 806:1
```

演習室 Python実行環境

■ 各個人が毎回ログインするたびに下を実行

```
$ cd /home/users/edu2019/lect/tk/

usetfというshellプログラムを探して下さい

$ bash usetf

$ source /tmp/tenst-tf/bin/activate
(test-rf) $
```

このコマンドプロンプトがでればOK
pythonの version 3が動いている

➔ novel/progに移動

文書分類/実習

■ SVMの学習

学習プログラム

```
$ cd /home/users/edu2019/lect/tk/nlsample/novel/prog
$ python svm_train.py
```

ただし，ここではエラーになります!!

```
[koichi@montecarlo prog]$ python svm_train.py
num_axis= 3010
shape (697, 3010)
acccuracy= 1.000
学習したモデルを保存 ../data/model/svm_model.dat
```

学習したモデル
を保存するため

注意

既に svm_model.datは置いています
もし自分で学習したい場合は，保存先を変更すること
次のページの testも中を見て変更して下さい

文書分類/実習

■ google colaboratory など演習室以外で行う場合

このnovelのディレクトリ全部をコピーすること

/home/users/edu2019/lect/tk/nlsample/novel

文書分類/実習

■ 学習結果の確認 (SVM)

評価プログラム

```
$ cd /home/users/edu2019/lect/tk/nlsample/novel/prog
$ python svm_test_analy.py
```

正解　　SVMの推定結果

```
2 2 m,「なかなか別品だったわねえ。
1 1 e,『何故です』って尋ねても、『まあいいから、そうしてお呉れな』と申して聞かな
                いのでございます。
2 2 m,もうおしまいになったじゃないか。
2 2 m,翌朝深淵の家へは医者が来たり、警部や巡査が来たりして、非常に雑※（「
                二点しんにょう＋鰥のつくり」、第4水準2-89-93）した。
1 1 e,数年以前から、いつもあんな苦し相な顔をして居ります。
0 0 a,五
0 1 a,そうしてこれが出来なければ、勿論二度とお父さんの所へも、帰れなくなるのに違いありません。
acccuracy= 0.903
```

精度 90.3%

文書分類/実習 (参考)

■ 学習結果の確認 (3層ニューラル)

学習プログラム

```
$ cd /home/users/edu2019/lect/tk/nlsample/novel/prog
$ python id_layer2_Bow.py (中身はLSTMなど混在してる良くない実装)
```

3層ニューラルの推定結果

```
final rate = 0.920000 (161/175)
rates [0.90857143 0.92      0.92      ]
0,0,婆さん は どこ から とる 出す た か 、 眼
1,1,私 は 仕方 が ない ので 母親 に 貰う た
2,2,小川 君 は 好奇 心 が 起る て 溜まる ない
0,0,ー そ 警察 へ 訴える う か ? 。 。
0,0,イツモダト 私 ハ 知 ラズ 知 ラズ 、 気 ガ
0,0,「 折角 御 嬢 さん の 在る か を つきとめる
```

正解

初期値の乱数により精度が変わる

ニューラルネットワークの場合は3回実行してベストのものを表示

精度 92%　(SVMを上回る)