

# Scikit-learn ゼミ

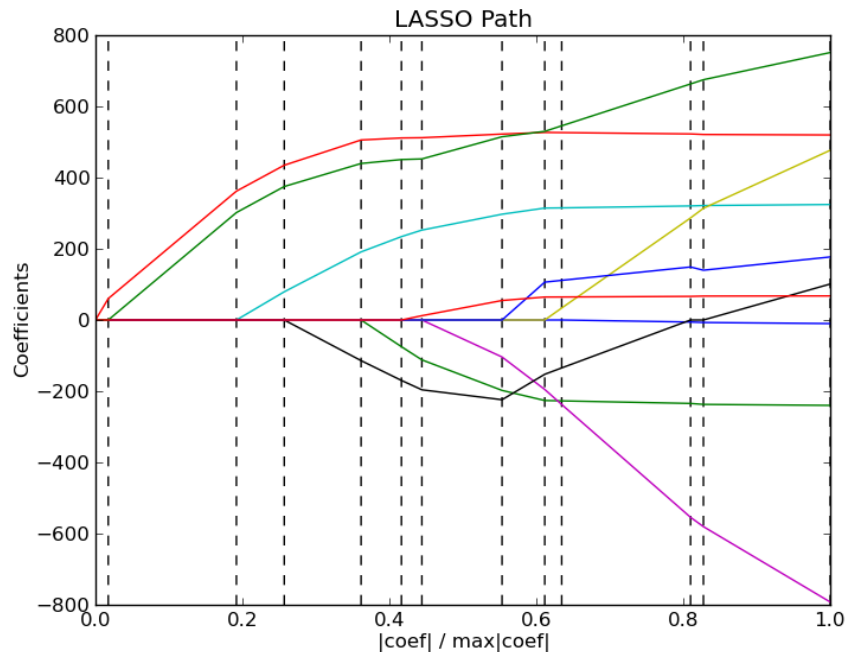
## 1.1.7 LARS Lasso

### 1.1.7.1 Mathematical formulation

河野和平

# LARS Lasso

- LARSを用いたLassoモデル
- 係数のノルムの関数で厳密解を得る
- 区分的に線形



# LARS Lasso

```
>>> from sklearn import linear_model
>>> clf = linear_model.LassoLars(alpha=.1)
>>> clf.fit([[0, 0], [1, 1]], [0, 1])
LassoLars(alpha=0.1, copy_X=True, eps=..., fit_intercept=True,
          fit_path=True, max_iter=500, normalize=True, precompute='auto',
          verbose=False)
>>> clf.coef_
array([ 0.717157...,  0.          ])
```

Python source code: `plot_lasso_lars.py`

```
print(__doc__)

# Author: Fabian Pedregosa <fabian.pedregosa@inria.fr>
#         Alexandre Gramfort <alexandre.gramfort@inria.fr>
# License: BSD 3 clause

import numpy as np
import pylab as pl

from sklearn import linear_model
from sklearn import datasets

diabetes = datasets.load_diabetes()
X = diabetes.data
y = diabetes.target

print("Computing regularization path using the LARS ...")
alphas, _, coefs = linear_model.lars_path(X, y, method='lasso', verbose=True)

xx = np.sum(np.abs(coefs.T), axis=1)
xx /= xx[-1]

pl.plot(xx, coefs.T)
ymin, ymax = pl.ylim()
pl.vlines(xx, ymin, ymax, linestyle='dashed')
pl.xlabel('|coef| / max|coef|')
pl.ylabel('Coefficients')
pl.title('LASSO Path')
pl.axis('tight')
pl.show()
```

# Mathematical formulation

- 各ステップで変数を有し, 残差と相関のある方向に増加する。
- `coef_path_`配列に格納される。
  - サイズ: `n_features` , `max_features+1`
  - 最初の列は常に0を取る。