Statistical Data Analysis
Discussion notes – week 6

- Problem sheet 3
- Comments on scikit-learn
- Comments on p-values

Problem sheet 3

Exercise 1 [6 marks]: Consider two continuous random variables x and y that follow the joint pdf $f(x, y)$ and define $u = x + y$. Show that the pdf of u can be written

$$g(u) = \int_{-\infty}^{\infty} f(x, u - x) \, dx \, .$$

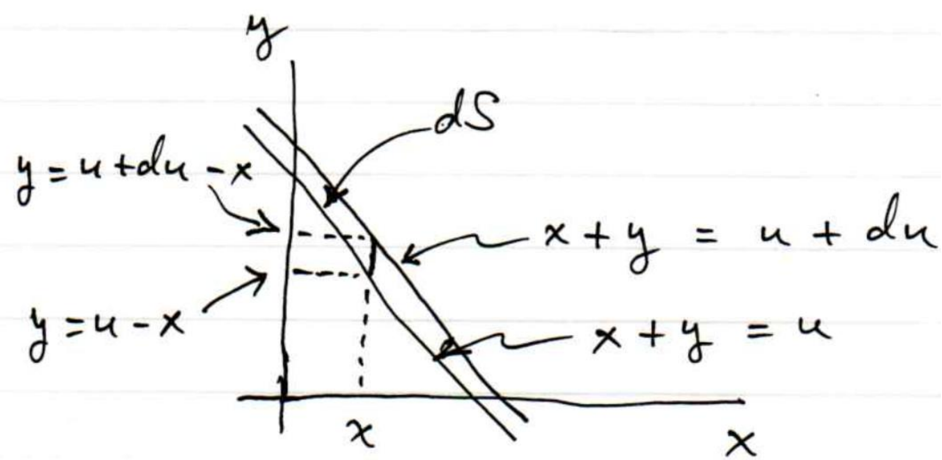
Use a method analogous to what was shown in the lectures for the product of two random variables (see p. 9 of the week 2 slides).

1) $x, y \sim f(x,y)$

Define $u = x + y$

From week 2 slides p. 9,

$$g(u)\,du = \int_{dS} f(x,y)\,dx\,dy$$



1 (cont.)

$$\Rightarrow \quad g(u)\,du = \int_{-\infty}^{\infty} \underbrace{\int_{u-x}^{u+du-x} f(x,y)\,dy}_{f(x,\,u-x)\cdot du} \, dx$$

(approx. const. in infintesimal interval)

(du ← interval width)

$$g(u)\,\cancel{du} = \int_{-\infty}^{\infty} f(x, u-x)\,dx\,\cancel{du}$$

QED

(or can carry out integration in opposite order,

$$g(u) = \int_{-\infty}^{\infty} f(u-y, y)\,dy$$)

2

Exercise 2 [7 marks]: Suppose x and y are independent and exponentially distributed each with mean values θ and define $u = x + y$. By using the result from Ex. 1, find the pdf of u. (In fixing the limits of integration, remember that the pdf is nonzero only for $x \geq 0$ and $y \geq 0$.)

$$x \sim \frac{1}{\xi} e^{-x/\xi} \, , \quad y \sim \frac{1}{\xi} e^{-y/\xi} \qquad \underline{\text{indep}}$$

$$\Rightarrow f(x,y) = \frac{1}{\xi^2} e^{-(x+y)/\xi} \, , \qquad \text{since } x, y \text{ indep.,} \quad f(x,y) = f_x(x) \, f_y(y)$$

$$u = x + y$$

$$g(u) = \int_{-\infty}^{\infty} f(x, u-x)\, dx$$

↑ nonzero for $x > 0$

and $y = u - x > 0$

$\Rightarrow$ $\underline{u > x}$

$$\Rightarrow g(u) = \int_0^u \frac{1}{\xi^2} e^{-(\cancel{x} + u - \cancel{x})/\xi}\, dx$$

$$= \frac{1}{\xi^2} e^{-u/\xi} \int_0^u dx$$

$$= \frac{u}{\xi^2} e^{-u/\xi} \qquad u \geq 0 .$$

3.

Exercise 3 [7 marks]: Consider a continuous random variable x that follows the pdf $f(x)$ with cumulative distribution $F(x)$, and suppose r follows a uniform distribution on $[0, 1]$. Prove (as was claimed in the lectures) that if we set $F(x) = r$ and solve for x, that $x(r)$ follows the pdf $f(x)$. To do this, use the method discussed in the lectures for finding the pdf of a function, and use the inverse function theorem, which says that

$$\frac{d}{dr} F^{-1}(r) = \frac{1}{\frac{dF}{dx}(x(r))} \, .$$

3) x follows $f(x)$

cdf is $F(x)$ $\Rightarrow$ $f(x) = \frac{dF}{dx}$

$r \sim$ Uniform$[0,1]$

i.e. $g(r) = 1$, $0 \le r \le 1$

If $F(x) = r$ $\Rightarrow$ $x = F^{-1}(r)$

pdf of $x(r)$ is

$$p(x) = g(r) \left| \frac{dr}{dx} \right| = \frac{g(r)}{\left| \frac{dx}{dr} \right|}$$

$$\frac{dx}{dr} = \frac{d}{dr} F^{-1}(r) = \frac{1}{\frac{dF}{dx}(x(r))} = \frac{1}{f(x(r))}$$

$$\Rightarrow \quad \frac{dr}{dx} = f(x)$$

$$\Rightarrow \quad p(x) = \underbrace{1}_{g(r)} \times \underbrace{f(x)}_{\left|\frac{dr}{dx}\right|} = \underline{f(x)}$$

Software for Machine Learning

We will practice ML with the Python package scikit-learn

`scikit-learn.org` ← software, docs, example code

scikit-learn built on NumPy, SciPy and matplotlib, so you need

```
import scipy as sp
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
```

and then you import the needed classifier(s), e.g.,

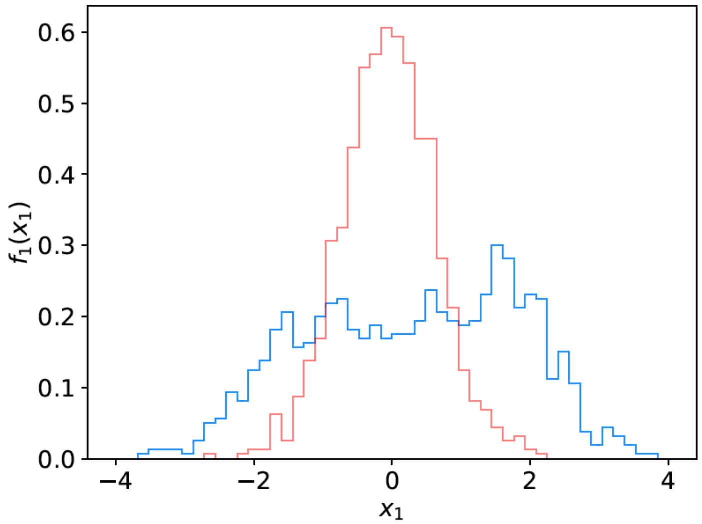
```
from sklearn.neural_network import MLPClassifier
```

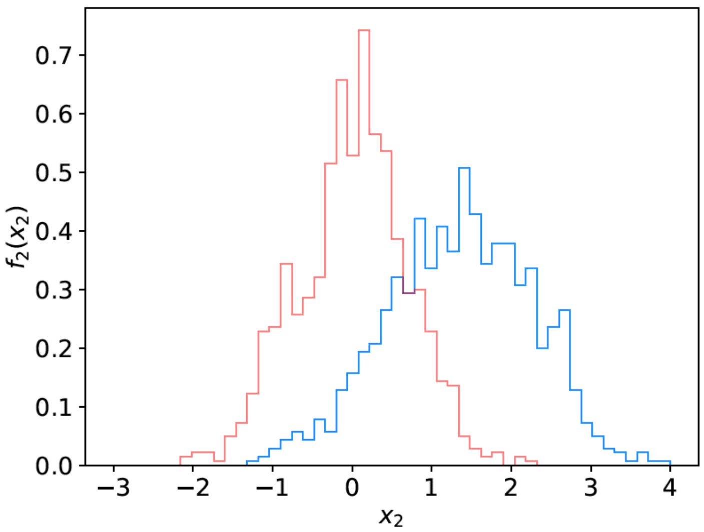
For a list of the various classifiers in scikit-learn see the docs on scikit-learn.org, also a very useful sample program:

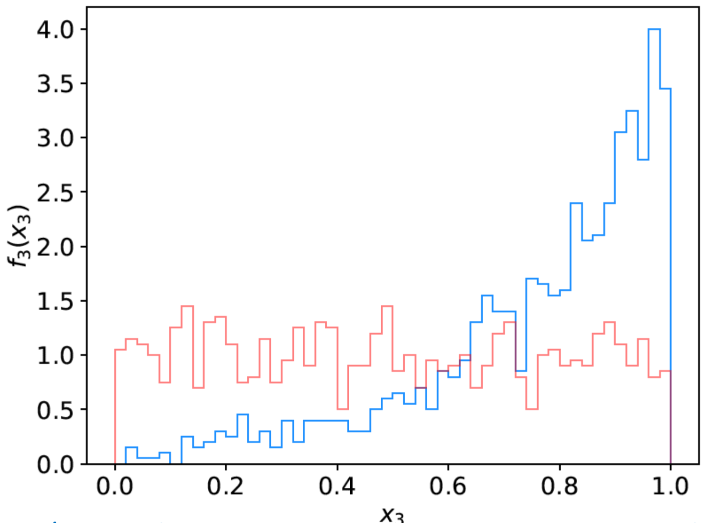
`http://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html`

Example: the data

We will do an example with data corresponding to events of two types: signal ($y = 1$, blue) and background ($y = 0$, red).







Each event is characterised by 3 quantities: $\boldsymbol{x} = (x_1, x_2, x_3)$.

Components are correlated.

Suppose we have 1000 events each of signal and background.

Reading in the data

scikit-learn wants the data in the form of numpy arrays:

```
#  read the data in from files,
# assign target values 1 for signal, 0 for background
sigData = np.loadtxt('signal.txt')
nSig = sigData.shape[0]
sigTargets = np.ones(nSig)
bkgData = np.loadtxt('background.txt')
nBkg = bkgData.shape[0]
bkgTargets = np.zeros(nBkg)

# concatenate arrays into data X and targets y
# split into two parts:  use one for training, the other for testing
X = np.concatenate((sigData,bkgData),0)
y = np.concatenate((sigTargets, bkgTargets))

# split data into training and testing samples
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5,
                                                    random_state=1)
```

Create, train, evaluate the classifier

Create an instance of the MLP (multilayer perceptron) class and “train”, i.e., adjust the values of the weights to minimise the loss function.

Here we request 3 hidden layers with 10 nodes each:

```
# create classifier object and train
clf = MLPClassifier(hidden_layer_sizes=(10,10,10), activation='tanh',
                    max_iter=2000, random_state=6)
clf.fit(X_train, y_train)
```

Use test data to see what fraction of events are correctly classified (default takes threshold of 0.5 for decision function)

```
# evaluate its accuracy (= 1 – error rate) using the test data
y_pred = clf.predict(X_test)
print(metrics.accuracy_score(y_test, y_pred))
```

Evaluating the decision function

So now for any point (x_1, x, x_3) in the feature space, we can evaluate the decision:

```
#  Test evaluation of decision function for a specific point in feature space
xt = np.array([0.37, 2.46, 0.42]).reshape((1,-1))
#t = clf.decision_function(xt)[0]          # not available for MLP
t = clf.predict_proba(xt)[0, 1]            # for MLP use this instead
```

Usually we have an array of points in $\boldsymbol{x}$-space, so we can get an array of probabilities:

```
t = clf.predict_proba(X_test)[:, 1]        # returns prob to be of type y=1
```

Can get this separately for the signal and background events and make histograms (see sample code).

Note for most other classifiers, the decision function is called decision_function – use this instead of predict_proba.

On defining a p-value

Earlier it was argued that the region of "equal or lesser compatibility" with H had greater compatibility with the predictions of some alternative hypothesis.

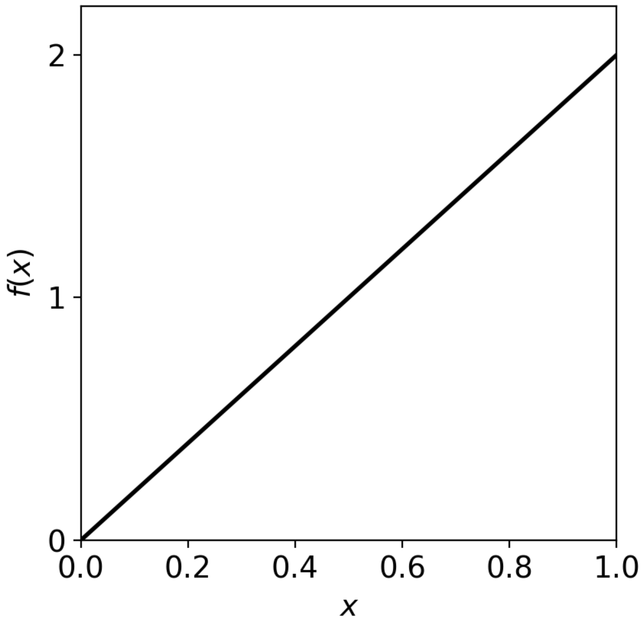
But shouldn't it be possible to identify such a region by using the pdf $f(x|H)$?

In general, no.

Consider cubic crystal grains produced by a process H that have a size distribution

$$f(x|H) = 2x, \quad 0 < x \leq 1$$

Observe grain of uncertain origin, measure x, find p-value of H.



If we observe a value x_{obs}, naively we could regard $x \leq x_{\mathrm{obs}}$ as constituting equal or less agreement with the predictions of $f(x|H)$.

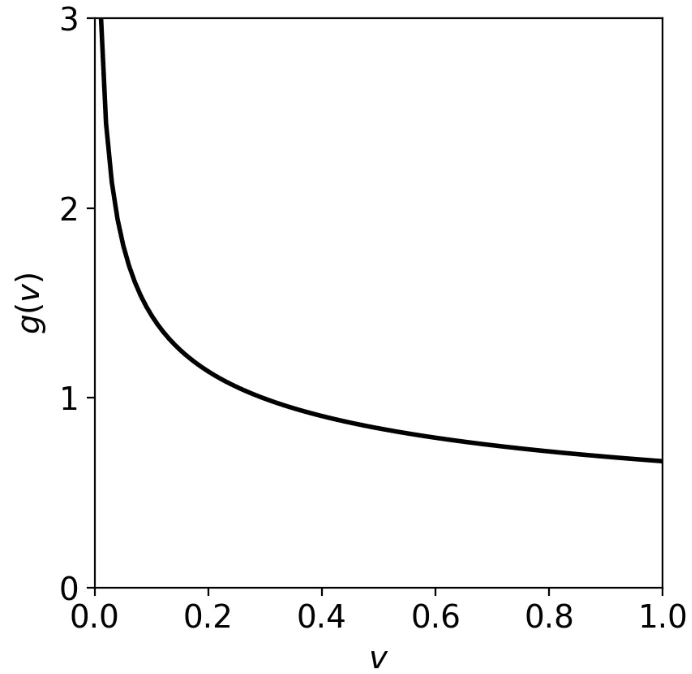
On defining a p-value (2)

But suppose we took the volume $v = x^3$ of the cube to represent its size. The volume distribution is

$$g(v|H) = f(x(v)|H) \left| \frac{dx}{dv} \right|$$

$$= \frac{2}{3} v^{-1/3}$$

$$0 < v \leq 1$$



So now it appears that smaller sizes are more compatible with H.

Conclusion: deciding what region of data space constitutes greater or lesser compatibility with *H* cannot be done by looking at the data distribution alone; it requires that one consider an alternative *H'*.

nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video | For A

Archive > Volume 519 > Issue 7541 > Research Highlights: Social Selection > Article

NATURE | RESEARCH HIGHLIGHTS: SOCIAL SELECTION

Psychology journal bans *P* values

Test for reliability of results 'too easy to pass', say editors.

Chris Woolston

26 February 2015 | Clarified: 09 March 2015

PDF | Rights & Permissions

A controversial statistical test has finally met its end, at least in one journal. Earlier this month, the editors of *Basic and Applied Social Psychology* (*BASP*) announced that the journal would no longer publish papers containing *P* values because the statistics were too often used to support lower-quality research [1].

https://imgs.xkcd.com/comics/significant.png