186

Mary Chris Go / 2014 11122

Activity 13: Expectation Maximization

Goal

To be able to overlay a plot of the estimated pdf on your fruit feature space

Expectation Algorithm

$$p(\boldsymbol{x}|\Theta)=\sum_{l=1}^{M} P_l\, p_l(\boldsymbol{x}|\boldsymbol{\theta}_l)$$

This kind of algorithm shows the distribution parameters. The sum of weight pdfs is the equation above whe **x** is the pdf of an ensemble and Θ is the set of parameters. P_l is the prior probability while p_l is the probability of observing **x** given the parameter.

This algorithm is being used to define the probability of an observation belonging to the lth distribution:

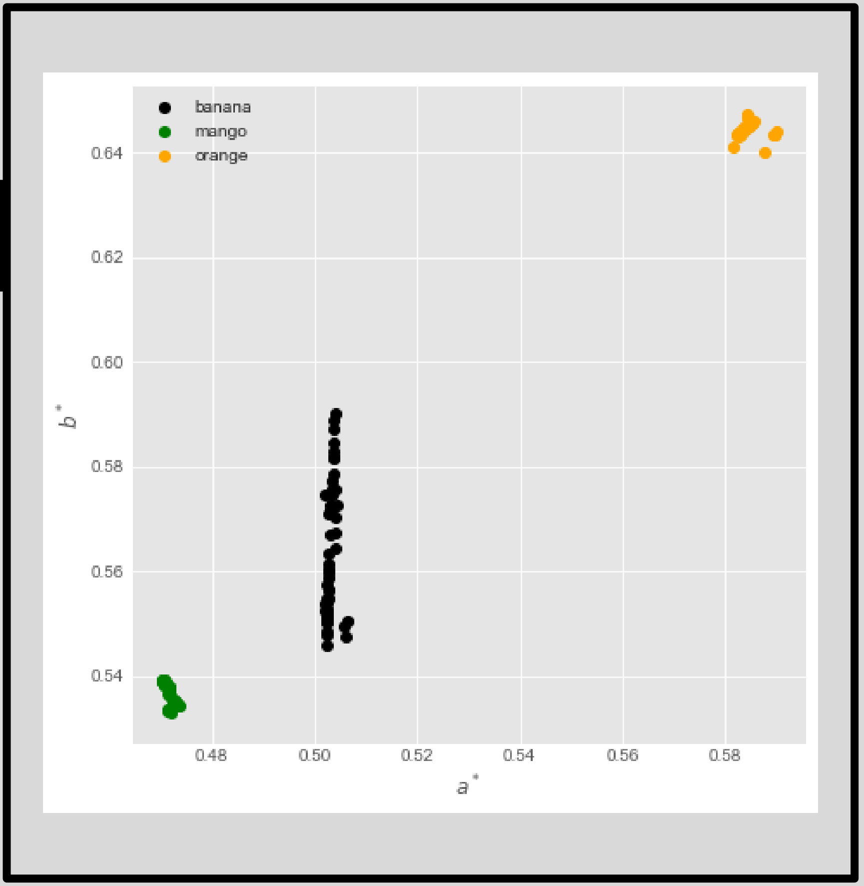
$$P(l|x_i,\boldsymbol{\Theta})=\frac{P_l\, p_l(x_i|\theta_l)}{p(x_i|\boldsymbol{\Theta})}=\frac{P_l\, p_l(x_i|\theta_l)}{\sum_{l=1}^{M} P_l\, p_l(x_i|\theta_l)}$$

For d-dimensional Gaussian distribution, component pdf is:

$$p_l(\boldsymbol{x}|\boldsymbol{\mu}_l,\boldsymbol{\Sigma}_l)=\frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_l|^{1/2}}\exp\{-1/2(\boldsymbol{x}-\boldsymbol{\mu}_l)^T\boldsymbol{\Sigma}_l^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_l)\}$$

Mary Chris Go / 2014 11122

Feature Space




```python
for j in range(3):
    filenames = os.listdir(dirs[j])
    for i,f in enumerate(filenames):
        if i == 50:
            break
        #eccentricity
        img = cv.imread(dirs[j] + f)
        img_gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
        thres, out = cv.threshold(img_gray, 127, 255, cv.THRESH_OTSU)
        out = (img_gray < thres).astype(float)
        img_label = meas.label(out)
        props = meas.regionprops(img_label)
        ecc = props[0]['eccentricity']

        # a* b*
        img_Lab = cv.cvtColor(img, cv.COLOR_BGR2Lab).astype(float)
        img_Lab /= img_Lab[:,:,0].max()
        img_L, img_a, img_b = cv.split(img_Lab)

        ass[j].append(img_a.mean())
        bss[j].append(img_b.mean())
        ecs[j].append(ecc)
```

This snippet of code was already used several times in clustering three kinds of fruit: orange, banana, and mango.

Algorithm

```python
def train(self, X, cluster_means, epochs=int(1e6)):
    self.graph_loss = []
    self.epochs = epochs
    N = len(X)
    ll_old = 0.
    self.theta = dict({'mu': cluster_means,
                       'cov': np.array([np.identity(self.dimensions) \
                                        for _ in range(self.n_classes)])
                      })
    self.P = np.ones(self.n_classes) * 1/self.n_classes
    for count in tnrange(epochs, desc='Epoch'):
        exp_A = []
        exp_B = []
        ll_new = 0.
        ws = np.zeros((self.n_classes, N))

        for l in range(self.n_classes):
            for i, x in enumerate(X):
                ws[l,i] = self.P[l] * \
                          stats.multivariate_normal(self.theta['mu'][l],
                                                    self.theta['cov'][l],
                                                   ).pdf(x)
        ws /= ws.sum(0)
```

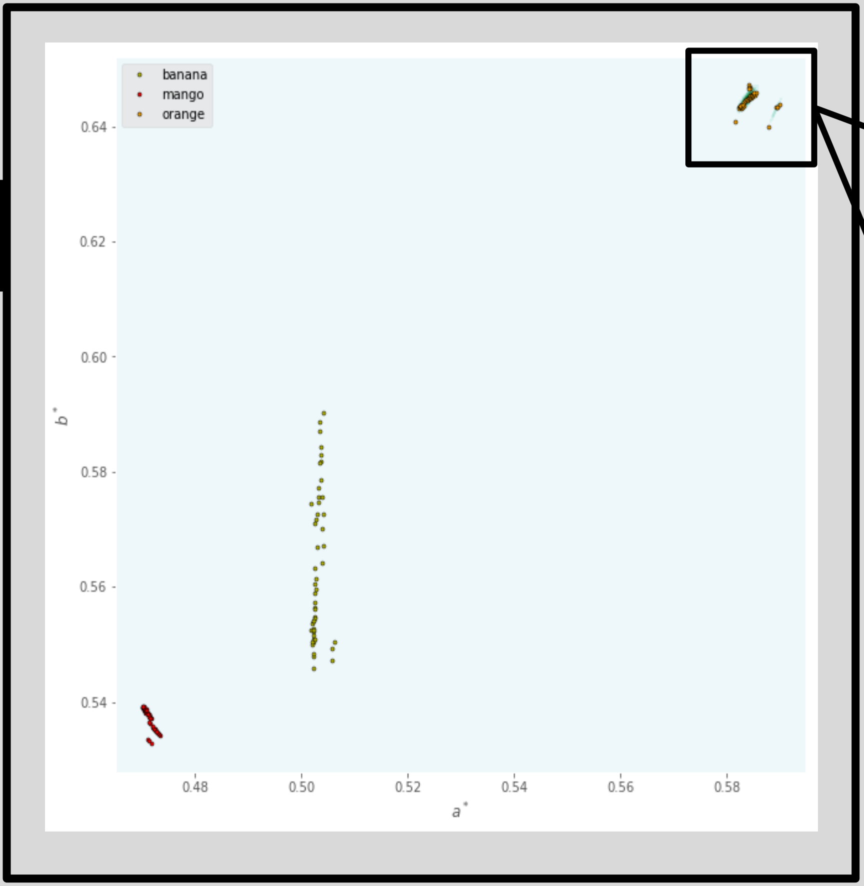
```
Epoch    0% 30/1000000 [00:06<53:51:13, 5.16it/s]
```

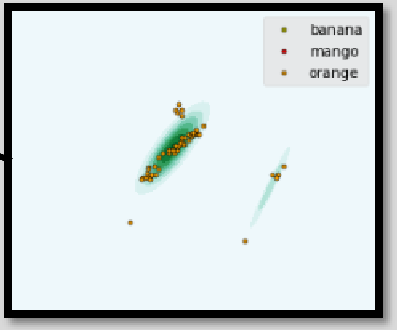
```python
xs = np.linspace(astars.min()-5e-3, astars.max()+5e-3, 5000)
ys = np.linspace(bstars.min()-5e-3, bstars.max()+5e-3, 5000)
```

Under class ExpectationMaximization is the def train that will try our machine. It will become easier right after this. We ‘concatenate’ three strings to append one string to the end of another. For this, we concatenate the three fruits for a and b. We stack them together through np.column_stacks.

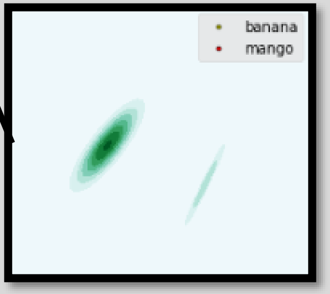
We apply np.meshgrid xs and xy then np.vstack the values from it. We can then plot the first column of features of the fruit versus the second column of it. We can superimposed the maximization and the points which will be show in the next slide.

Expectation Maximization





Close up portion of the graph where the algorithm take into effect.



The maximization without the points.

The algorithm showed the expected to the orange fruit. The probabilities were shown. Wha’ts amazing is the points were mostly in the distribution calculated. Somehow, it wasn’t showing anything for the mango and banana. This mistake can be blame to the algorithm I made. ☹

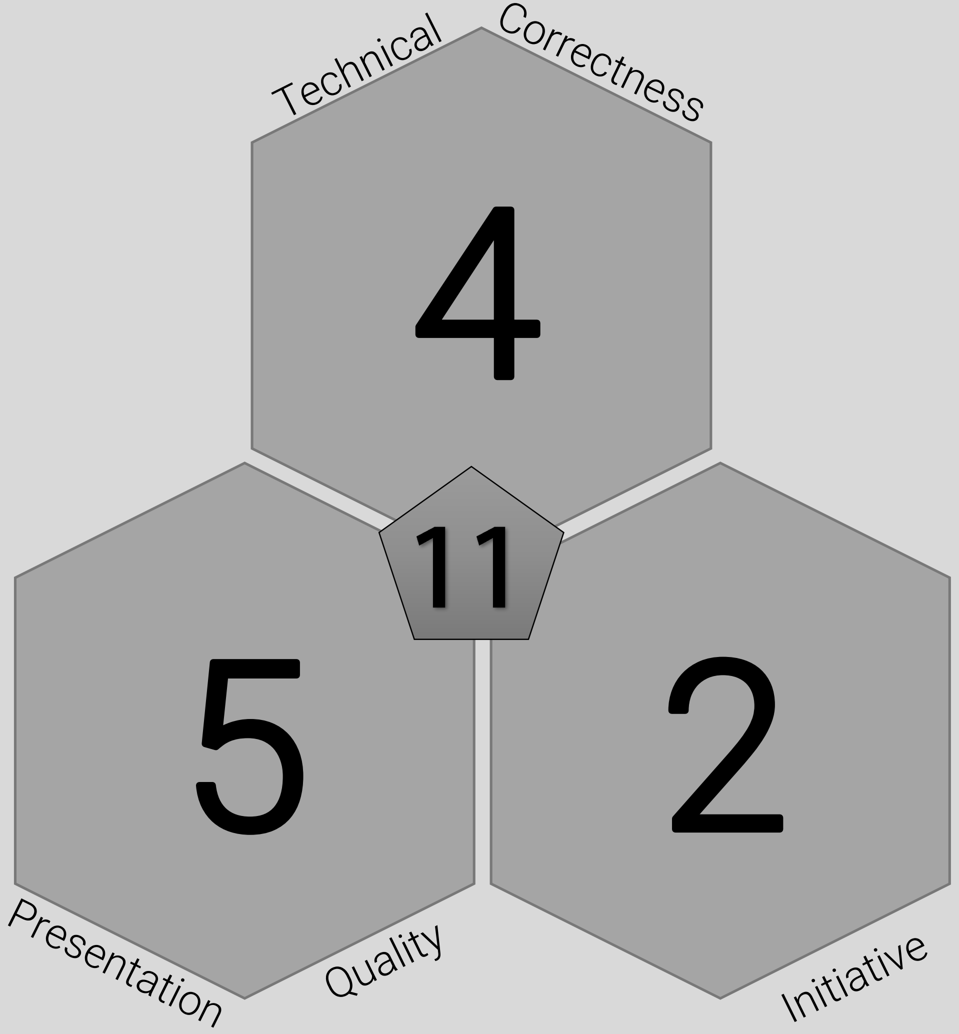
Mary Chris Go / 2014 11122

Summary

I was able to pull it off but only for one fruit. It is still a mystery to me as to why it's not showing for the other fruits. This problem is something to look in the future to learn more about the topic also. Thank you to some of my classmates who helped me along the way.

All in all, it was frustrating to do it. But since there are two deadlines to beat for 186 this week, I can't give full time attention to this. Though it was a good learning.

Self-Evaluation



Mary Chris Go / 2014 11122

References

- Soriano, M., “Expectation Maximization”. 2019