

Homework 4 - Problem 8

```
In [ ]: import numpy as np
import seaborn as sns
from pylab import rcParams
from sklearn.manifold import TSNE
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
```

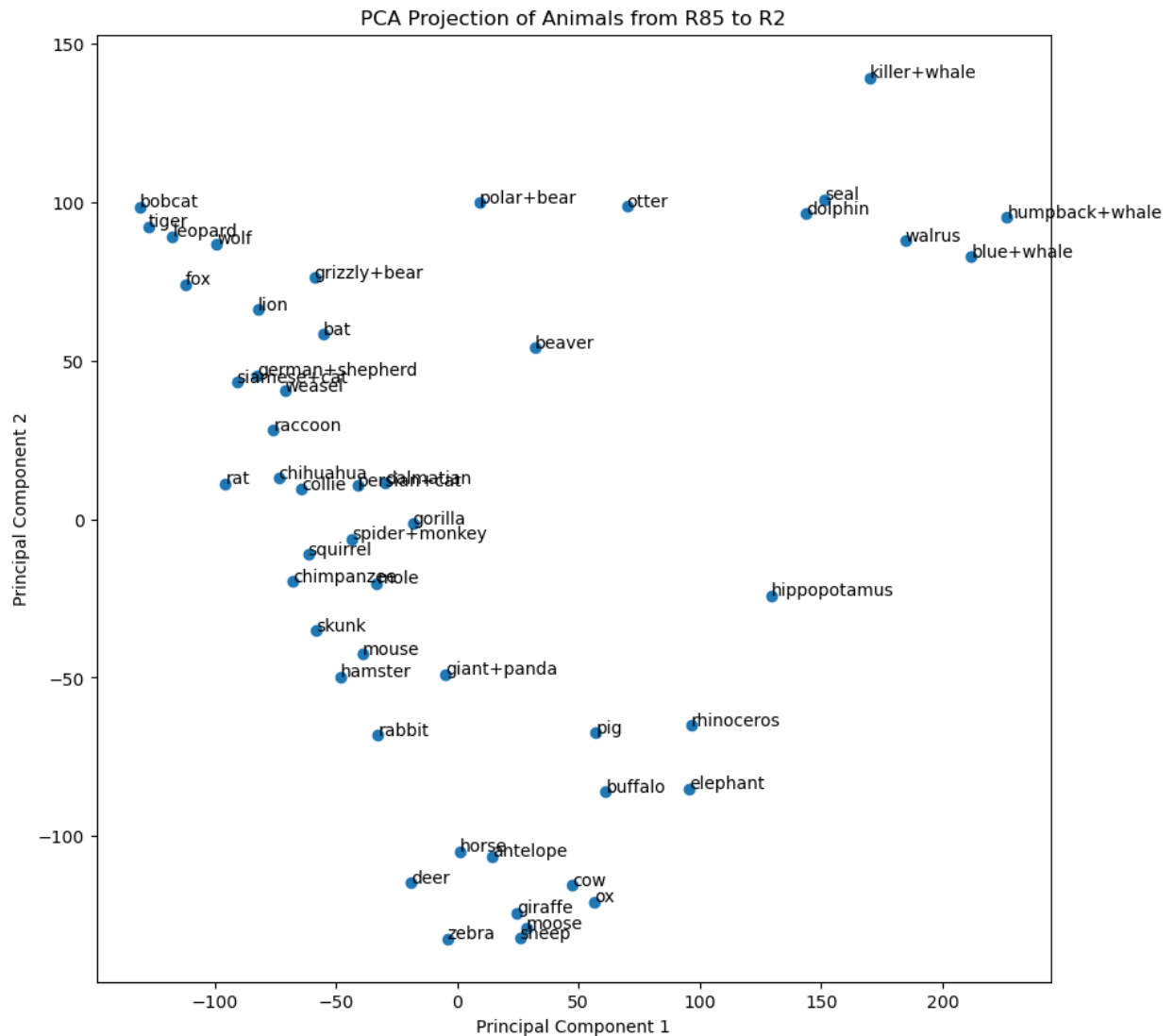
```
In [2]: real_val = np.loadtxt('predicate-matrix-continuous.txt')
names = np.genfromtxt('classes.txt', dtype=str)
```

```
In [15]: animal_names = names[:,1]
```

```
In [4]: pca = PCA(n_components = 2)
pca = pca.fit_transform(real_val)
```

```
In [19]: #plotting
rcParams['figure.figsize'] = 10,10
plt.scatter(pca[:,0], pca[:,1])
for i, name in enumerate(animal_names):
    plt.annotate(name, (pca[i, 0], pca[i, 1]))
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.title('PCA Projection of Animals from R85 to R2')
```

Out[19]: Text(0.5, 1.0, 'PCA Projection of Animals from R85 to R2')



b)

In [83]:

```

perplxts = [5, 10, 25, 50]
rcParams['figure.figsize'] = 15, 15
fig, axes = plt.subplots(2, 2)
tsne_models=[]

for i, per in enumerate(perplxts) :
    model = TSNE(n_components = 2, perplexity = per).fit_transform(re
    tsne_models.append(model)

    ax = axes[i // 2 , i % 2]
    ax.scatter(model[:,0], model[:,1])
    ax.set_title(f't-SNE Perplexity: {per}')
    for j, name in enumerate(animal_names):
        ax.annotate(name, (model[j,0],model[j,1]))

```

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:780: FutureWarning: The default initialization in TSNE will change from 'random' to 'pca' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:790: FutureWarning: The default learning rate in TSNE will change from 200.0 to 'auto' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:780: FutureWarning: The default initialization in TSNE will change from 'random' to 'pca' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:790: FutureWarning: The default learning rate in TSNE will change from 200.0 to 'auto' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:780: FutureWarning: The default initialization in TSNE will change from 'random' to 'pca' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:790: FutureWarning: The default learning rate in TSNE will change from 200.0 to 'auto' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:780: FutureWarning: The default initialization in TSNE will change from 'random' to 'pca' in 1.2.

warnings.warn(

/Users/marlenearredondo/opt/anaconda3/lib/python3.9/site-packages/sklearn/manifold/_t_sne.py:790: FutureWarning: The default learning rate in TSNE will change from 200.0 to 'auto' in 1.2.

warnings.warn(





```
In [108]: def avg_distortion(original_D, embedded_D):
            c = (np.mean(original_D)) / (np.mean(embedded_D)) #scaling factor
            avg_dist = np.maximum(original_D / (c*embedded_D), (c*embedded_D)/
            avg_dist[np.isnan(avg_dist) | np.isinf(avg_dist)] = 1 #ensure it i
            avg_dist = np.mean(avg_dist)
            return avg_dist
```

```
In [109]: #original distances
dim = real_val.shape[0]
originaldist = np.zeros((dim,dim))

for i in range(dim):
    for j in range(i+1, dim):
        originaldist[i,j] = originaldist[j,i] = euclidean_distance(real_val[i], real_val[j])
```

```
In [110]: #embedded distances - pca
dim1 = pca.shape[0]
pca_dist = np.zeros((dim,dim))

for i in range(dim1):
    for j in range(i + 1, dim1):
        pca_dist[i, j] = pca_dist[j, i] = euclidean_distance(pca[i], pca[j])
```

```
In [116]: #embedded distances - t-SNE
dim2 = model.shape[0]
tsne_dists = [np.zeros((dim2, dim2)) for _ in range(len(tsne_models))]

for idx, tsne_model in enumerate(tsne_models):
    for i in range(dim2):
        for j in range(i + 1, dim2):
            tsne_dists[idx][i, j] = tsne_dists[idx][j, i] = euclidean_distance(tsne_model[i], tsne_model[j])
```

```
In [117]: avg_distpca = avg_distortion(originaldist, pca_dist)
avg_disttsne = [avg_distortion(originaldist, d) for d in tsne_dists]

/var/folders/rp/mtyztnnd6p35kq9b5_xbdn7c0000gn/T/ipykernel_52470/626396907.py:3: RuntimeWarning: invalid value encountered in true_divide
  avg_dist = np.maximum(original_D / (c*embedded_D), (c*embedded_D)/original_D)
```

```
In [118]: print(avg_distpca)

1.8147225492939218
```

```
In [119]: avg_disttsne
```

```
Out[119]: [2.6946228062252406,
 1.6163054789842888,
 1.4544118386842453,
 1.6916977021518795]
```