CPSC 340 Assignment 5 (due Friday March 23 at 9:00pm)

1 MAP Estimation

$$f(w) = \frac{1}{2}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2.$$

For each of the alternate assumptions below, show how the loss function would change (simplifying as much as possible):

1. We use a zero-mean Laplace prior for each variable with a scale parameter of λ^{-1}, so that

$$p(w_j) = \frac{\lambda}{2}\exp(-\lambda|w_j|).$$

We're only changing the prior, so the beginning part of the function stays the same ($\frac{1}{2}\|Xw - y\|^2$).
If we take the negative log of $\frac{\lambda}{2}\exp(-\lambda|w_j|)$, it becomes $-ln(\frac{\lambda}{2}) + \lambda|w_j|$ for a given j.
Hence, for all j, we get $-ln(\frac{\lambda}{2}) + \sum_{j=1}^{n}|w_j|$.

Thus, the change is that we end up using L1-regularization instead of L2-regularization.

2. We use a Laplace likelihood with a mean of $w^T x_i$ and a scale of 1, so that

$$p(y_i|x_i, w) = \frac{1}{2}\exp(-|w^T x_i - y_i|).$$

We're only changing the likelihood, so the end part of the function stays the same ($\frac{\lambda}{2}\|w\|^2$).
If we take the negative log of $\frac{1}{2}\exp(-|w^T x_i - y_i|)$, it becomes $-ln(\frac{1}{2}) + |w^T x_i - y|$ for a given i.
Hence, for all i, we get $-ln(\frac{1}{2}) + \sum_{i=1}^{n}|w^T x_i - y_i|$, which is equivalent to $-ln(\frac{1}{2}) + |Xw - y|_1$.

Thus, the change is that our data fitting term now uses the L1 norm.

3. We use a Gaussian likelihood where each datapoint has variance σ^2 instead of 1,

$$p(y_i|x_i, w) = \frac{1}{\sqrt{2\sigma^2\pi}}\exp\left(-\frac{(w^T x_i - y_i)^2}{2\sigma^2}\right).$$

We're only changing the likelihood, so the end part of the function stays the same ($\frac{\lambda}{2}\|w\|^2$).
If we take the negative log of $p(y_i|x_i, w) = \frac{1}{\sqrt{2\sigma^2\pi}}\exp\left(-\frac{(w^T x_i - y_i)^2}{2\sigma^2}\right)$, we get $-ln(\frac{1}{\sqrt{2\sigma^2\pi}} + \frac{(w^T x_i - y_i)^2}{2\sigma^2}$ for a given i.
Hence, for all i, we get $-ln(\frac{1}{\sqrt{2\sigma^2\pi}}) + \sum_{i=1}^{n}\frac{(w^T x_i - y_i)^2}{2\sigma^2}$, which is equivalent to $-ln(\frac{1}{\sqrt{2\sigma^2\pi}} + \frac{||Xw - y||_2}{2\sigma^2}$.

Thus, the change is that our data fitting term is also divided by the variance.

4. We use a Gaussian likelihood where each datapoint has its own variance σ_i^2,

$$p(y_i|x_i, w) = \frac{1}{\sqrt{2\sigma_i^2\pi}}\exp\left(-\frac{(w^T x_i - y_i)^2}{2\sigma_i^2}\right).$$

We're only changing the likelihood, so the end part of the function stays the same ($\frac{\lambda}{2}\|w\|^2$).
If we take the negative log of $p(y_i|x_i, w) = \frac{1}{\sqrt{2\sigma_i^2\pi}} \exp\left(-\frac{(w^T x_i - y_i)^2}{2\sigma_i^2}\right)$, we get $-ln(\frac{1}{\sqrt{2\sigma_i^2\pi}} + \frac{(w^T x_i - y_i)^2}{2\sigma^2}$ for a given i.
Hence, for all i, we get $-ln(\frac{1}{\sqrt{2\sigma_i^2\pi}}) + \sum_{i=1}^{n} \frac{(w^T x_i - y_i)^2}{2\sigma^2}$.
Let S be a matrix where values of $\frac{1}{2\sigma_1^2}$ to $\frac{1}{2\sigma_i^2}$ run along the main diagonal. We can then simplify the above equation to: $-ln(\frac{1}{\sqrt{2\sigma_i^2\pi}}) + (Xw - y)^T A(Xw - y)$.

Thus, the change is that we have n additional $sigma^2$ that divides the L2 norm of our data fitting term.

2 Principal Component Analysis

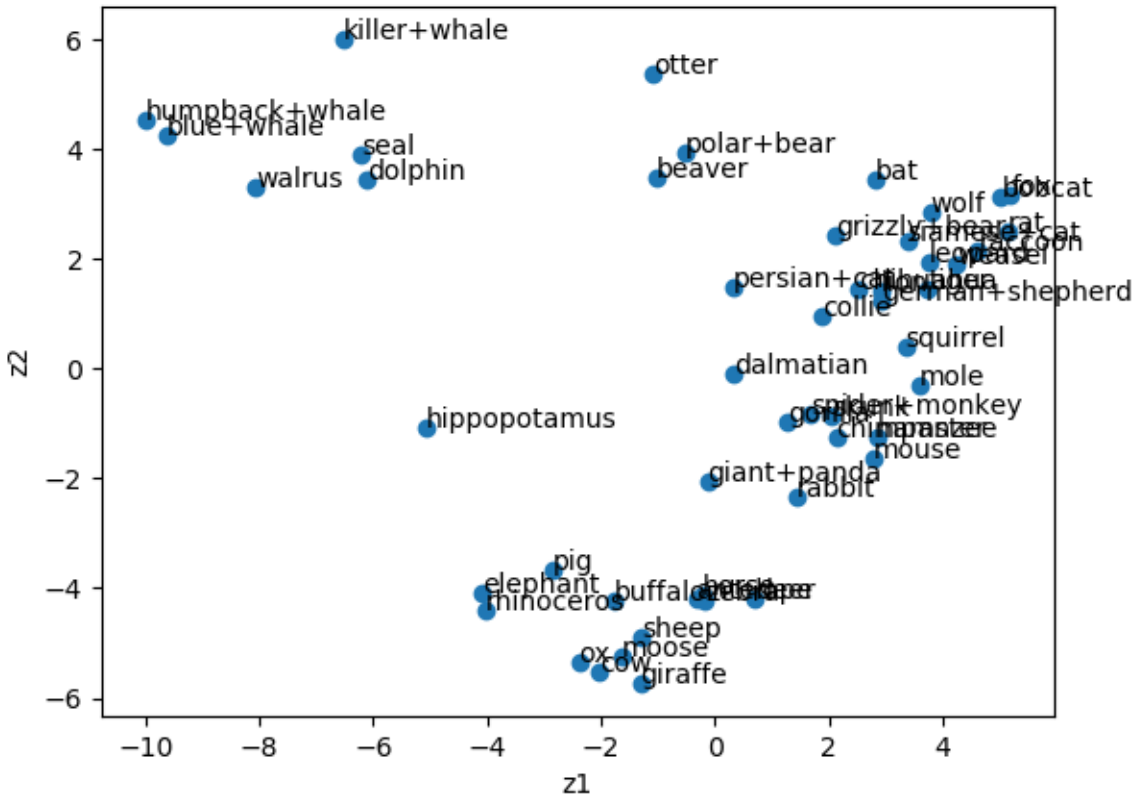
2.1 PCA by Hand

1. What is the first principal component?
   The first principle component is $w_1 = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$.
2. What is the (L2-norm) reconstruction error of the point (3,3)? (Show your work.)
   First, we want to get z by subtracting by the means and multiplying by our principal component.
   $$z = (3-0)(\frac{1}{\sqrt{2}}) + (3-1)(\frac{1}{\sqrt{2}}) = \frac{5}{\sqrt{2}}$$
   .
   Then, we want to get $\hat{x}$ by multiplying by our principal component and add back the means.
   $$z = \frac{5}{\sqrt{2}}(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}) + (0,1) = (2.5, 3.5)$$
   We can get the reconstruction error like so:
   $$\sqrt{(2.5-3)^2 + (3.5-3)^2}$$
   $$= \frac{1}{\sqrt{2}}$$
3. What is the (L2-norm) reconstruction error of the point (3,4)? (Show your work.)
   First, we want to get z by subtracting by the means and multiplying by our principal component.
   $$z = (3-0)(\frac{1}{\sqrt{2}}) + (4-1)(\frac{1}{\sqrt{2}}) = \frac{6}{\sqrt{2}}$$
   .
   Then, we want to get $\hat{x}$ by multiplying by our principal component and add back the means.
   $$z = \frac{6}{\sqrt{2}}(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}) + (0,1) = (3,4)$$
   We can get the reconstruction error like so:
   $$\sqrt{(3-3)^2 + (4-4)^2}$$
   $$= 0$$

2.2 Data Visualization

Hand in your code and the scatterplot.

The code can be found at *../code/main.py*.



2.3 Data Compression

1. How much of the variance is explained by our 2-dimensional representation from the previous question?
   30.19% of the variance is explained.
2. How many PCs are required to explain 50% of the variance in the data?
   We need 5 or more PCs to explain 50% of the variance.

3 PCA Generalizations

3.1 Robust PCA

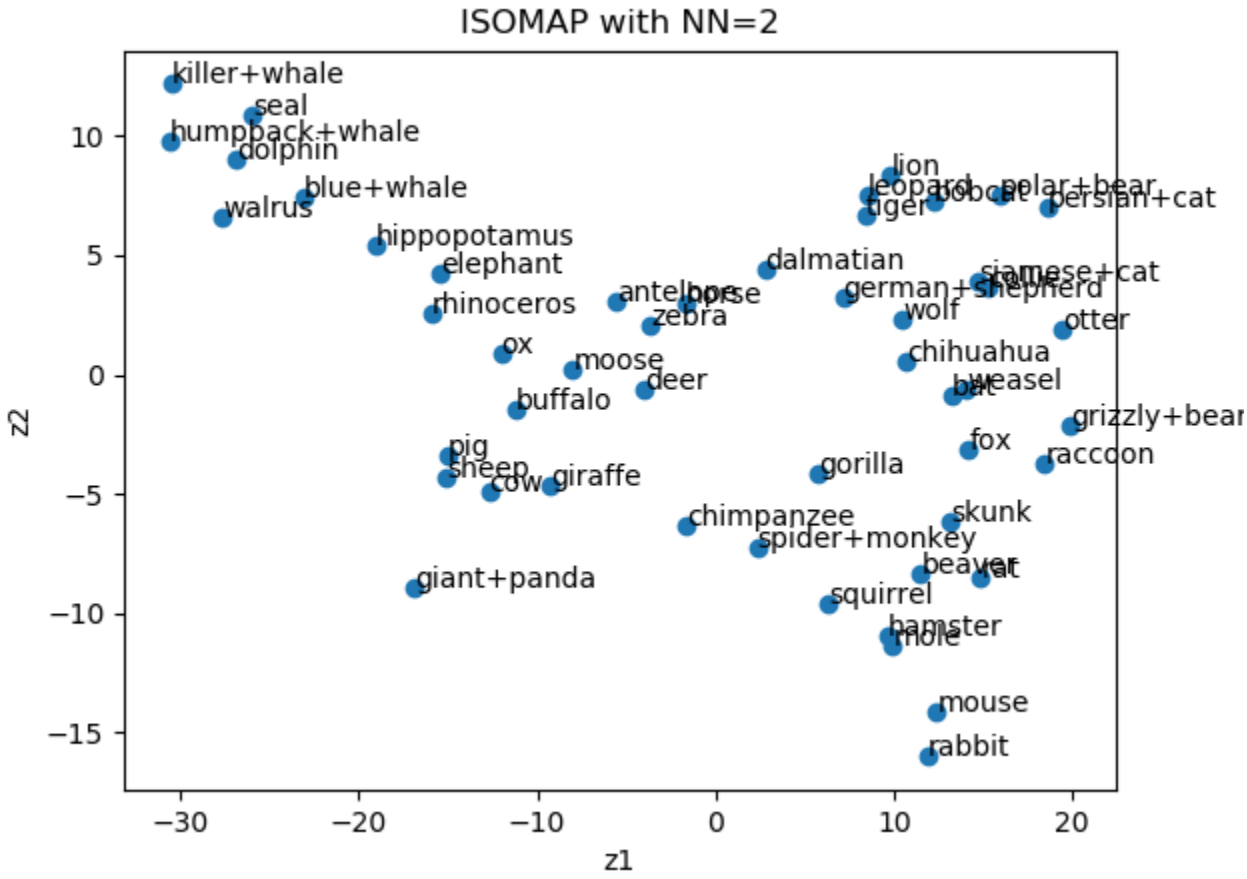
Complete the class *pca.RobustPCA*, that uses a smooth approximation to the absolute value to implement robust PCA. Comment on the quality of the results.
The code can be found at *../code/pca.py*.

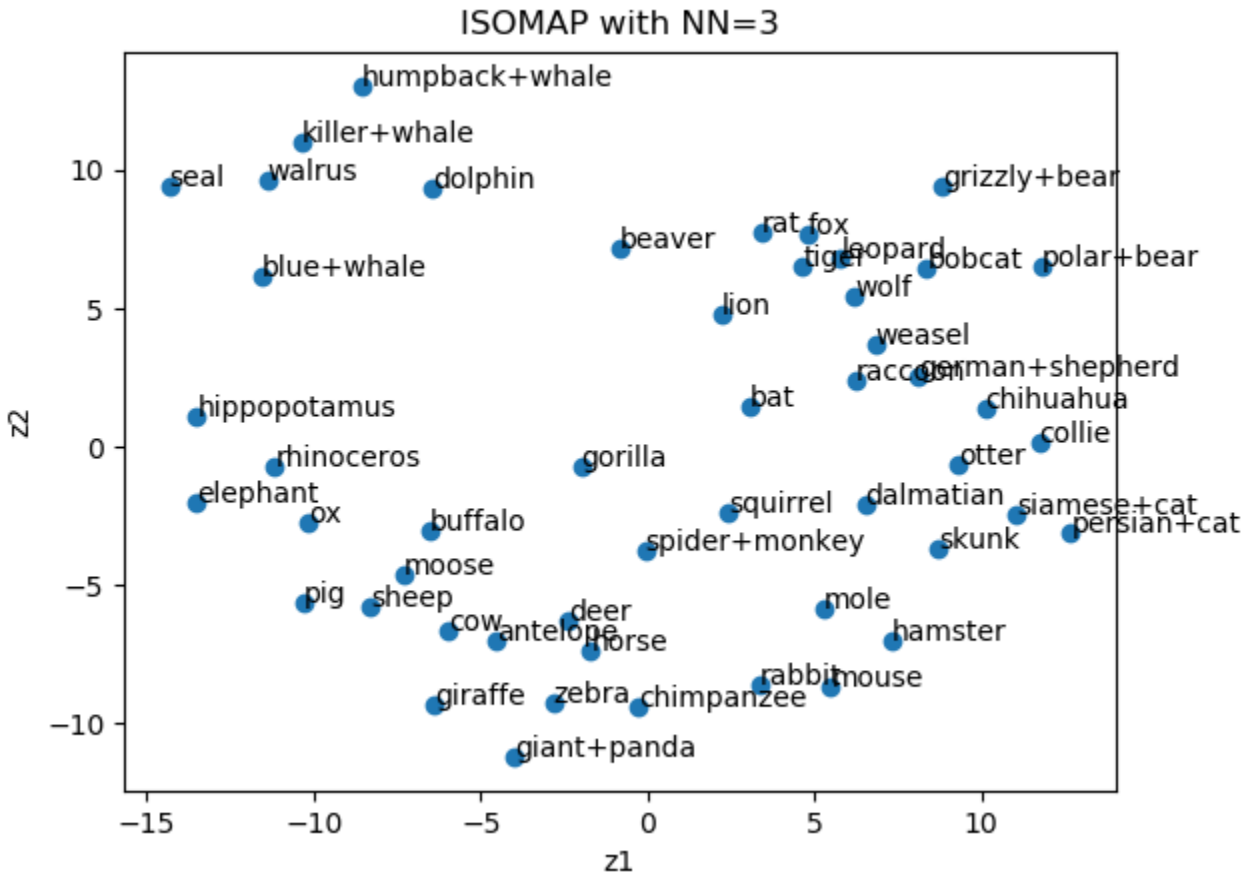
The results look much better, it identifies the cars on the highway a lot better than the old version does, and the blobs of the cars moving on the highway are reduced in the background subtraction bit.

4 Multi-Dimensional Scaling

4.1 ISOMAP

Plot the results using 2 and using 3-nearest neighbours. The code can be found at *../code/manifold.py*.





4.2 Reflection

Briefly comment on PCA vs. MDS vs. ISOMAP for dimensionality reduction on this particular data set. In your opinion, which method did the best job and why?
ISOMAP with k = 2 seems to do the best job as it provides individual clumpings without vacuuming all the animals to one place like PCA does. MDS and ISOMAP with k = 3 don't seem to give very clear groupings of the animals and seems very general.

5 Very-Short Answer Questions

1. Why is the kernel trick often better than explicitly transforming your features into a new space?
   Because you dont have to store the Z and w, uses the dot product between all the training examples instead of storing the training examples themselves. Some ks are just too large to store makes multi-dimensional polynomial basis intractable.
2. Why is the kernel trick more popular for SVMs than with logistic regression?
   Kernel trick would have to process all the points in the logistic regression case, while it would only

need to process the support vector examples in the case of the SVM. so it would be way less time consuming.

3. What is the key advantage of stochastic gradient methods over gradient descent methods?
   Iterations are n times faster than gradient descent iterations because it uses the gradient of a randomly chosen training example.

4. Does stochastic gradient descent with a fixed α converge to the minimum of a convex function in general?
   No, because as the points reach the confusion point nearer the min the direction of the gradient becomes confused and goes in many directions. So if you left it as the same size it would just stay in this ball (with a radius of α^t).

5. What is the difference between multi-label and multi-class classification?
   In multi-class there is one label vs. multi-label = many labels where many can be correct.

6. What is the difference between MLE and MAP?
   For MLE, we're looking for the maximizer that maximizes the likelihood. For MAP, we're looking for the maximizer that maximizes likelihood multiplied by the prior.

7. Linear regression with one feature and PCA with 2 features (and $k = 1$) both find a line in a two-dimensional space. Do they find the same line? Briefly justify your answer.
   No. Linear regression tries to find a line minimizing vertical squared distance (we only care about predicting the other feature), while PCA tries to find a line that minimizes squared distance in both directions.

8. Are the vectors minimizing the PCA objective unique? Briefly justify your answer.
   No. They are non-unique due to reasons such as ordering, scaling and rotation.

9. Name two methods for promoting sparse solutions in a linear regression model that result in convex problems.
   Non-negative least squares, and L1 regularization.

10. Can we use the normal equations to solve non-negative least squares problems?
    No. Solving for least squares and then setting all negative weights to 0 would be a naive solution, as you may no longer be at the best solution after setting your negative weights to 0.