Name:

Giovanni Zavalza

Netid:

gzava3

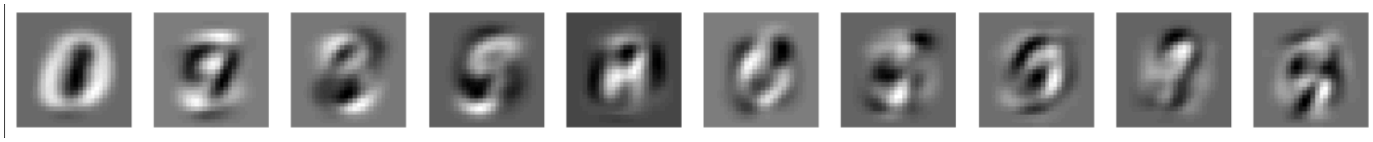
CS 441 - HW2: PCA and Linear Models

Complete the sections below. You do not need to fill out the checklist.

Total Points Available	**[ ] / 160**
1. PCA on MNIST	
a. Display 10 principal component vectors	[ ] / 5
b. Display scatterplot	[ ] / 5
c. Plot cumulative explained variance	[ ] / 5
d. Compression and 1-NN experiment	[ ] / 15
2. MNIST Classification with Linear Models	
a. LLR / SVM error vs training size	[ ] / 20
b. Error visualization	[ ] / 10
c. Parameter selection experiments	[ ] / 15
3. Temperature Regression	
a. Linear regression test	[ ] / 10
b. Feature selection results	[ ] / 15
4. Stretch Goals	
a. PR and ROC curves	[ ] / 10
b. Visualize weights	[ ] / 10
c. Other embeddings	[ ] / 15
d. One city is all you need	[ ] / 15
e. SVM with RBF kernel	[ ] / 10

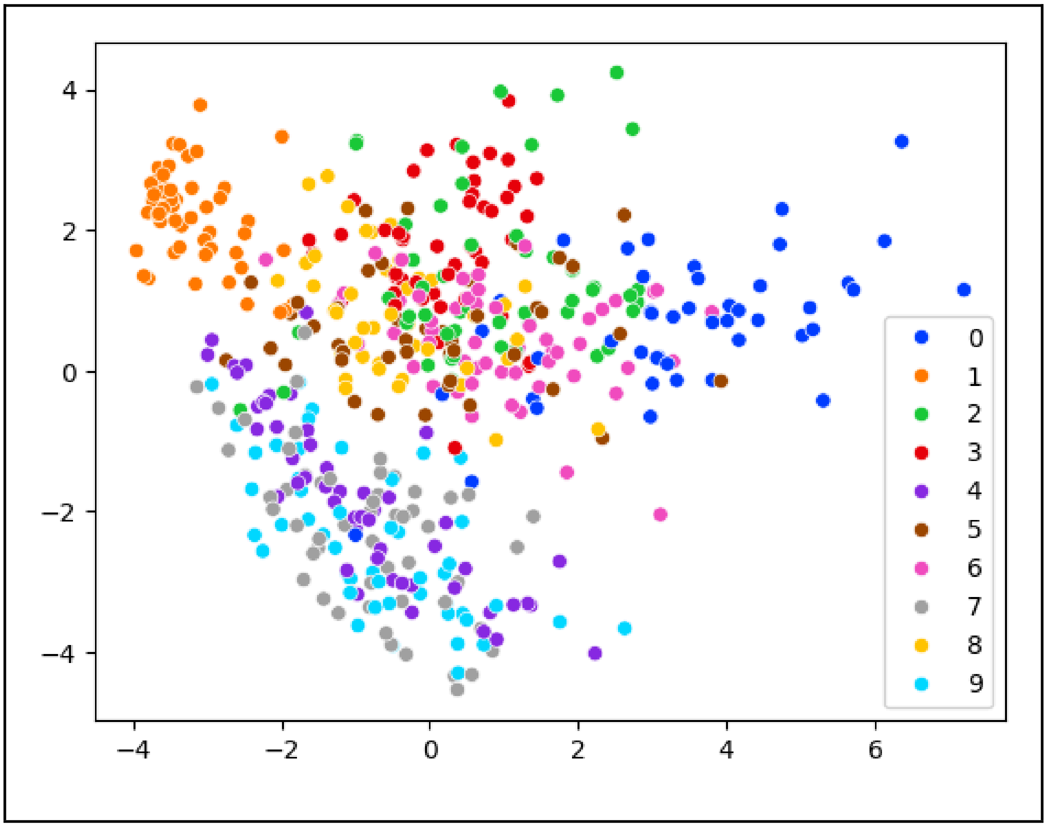
1. PCA on MNIST

a. Display 10 principal component vectors

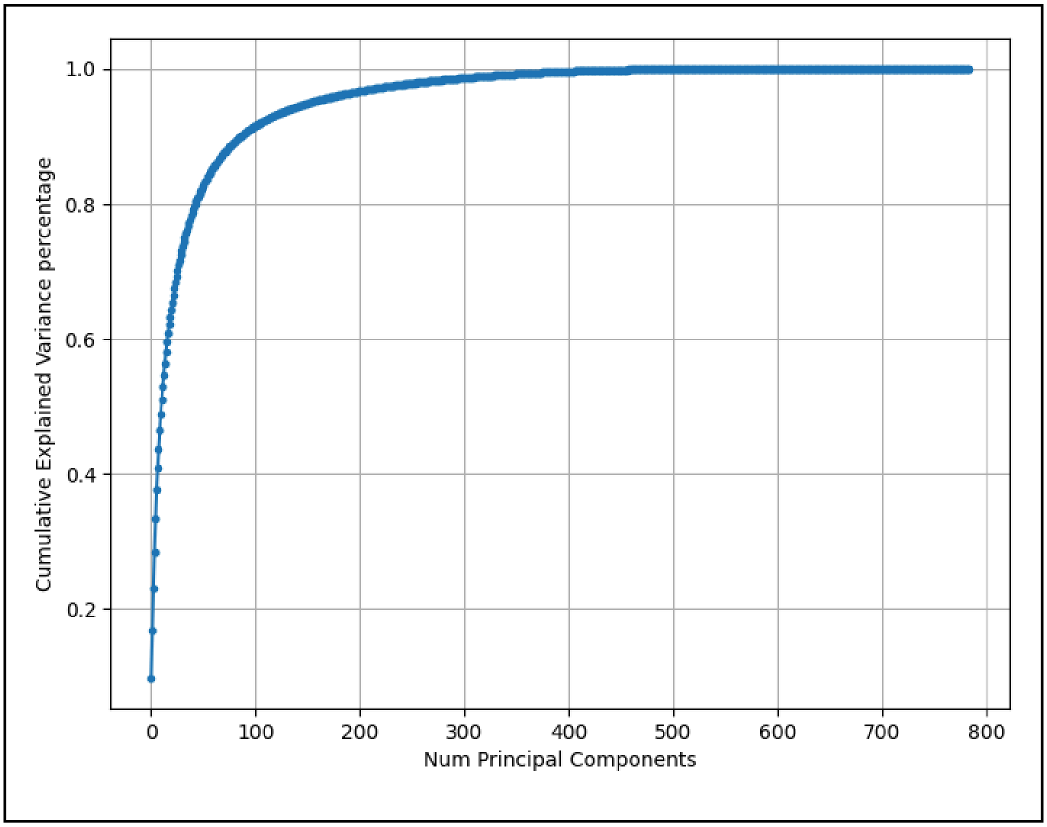


b. Display scatterplot

Scatterplot `x_train[:500]` for the first two PCA dimensions. Show a different color for each label.



c. Plot cumulative explained variance



d. Compression and 1-NN experiment

Number of components selected

	Total Time (s)	Test Error (%)[1]	Dimensions[2]
Brute Force (PCA)	3.27s	2.66%	87
Brute Force	30.52s	3.09%	784

1. Test error for PCA should be lower than non-PCA in this case.
2. Dimensions should be somewhere in the range of 50-100.

2. MNIST Classification with Linear Models

a. LLR / SVM error vs training size

Test error (%)

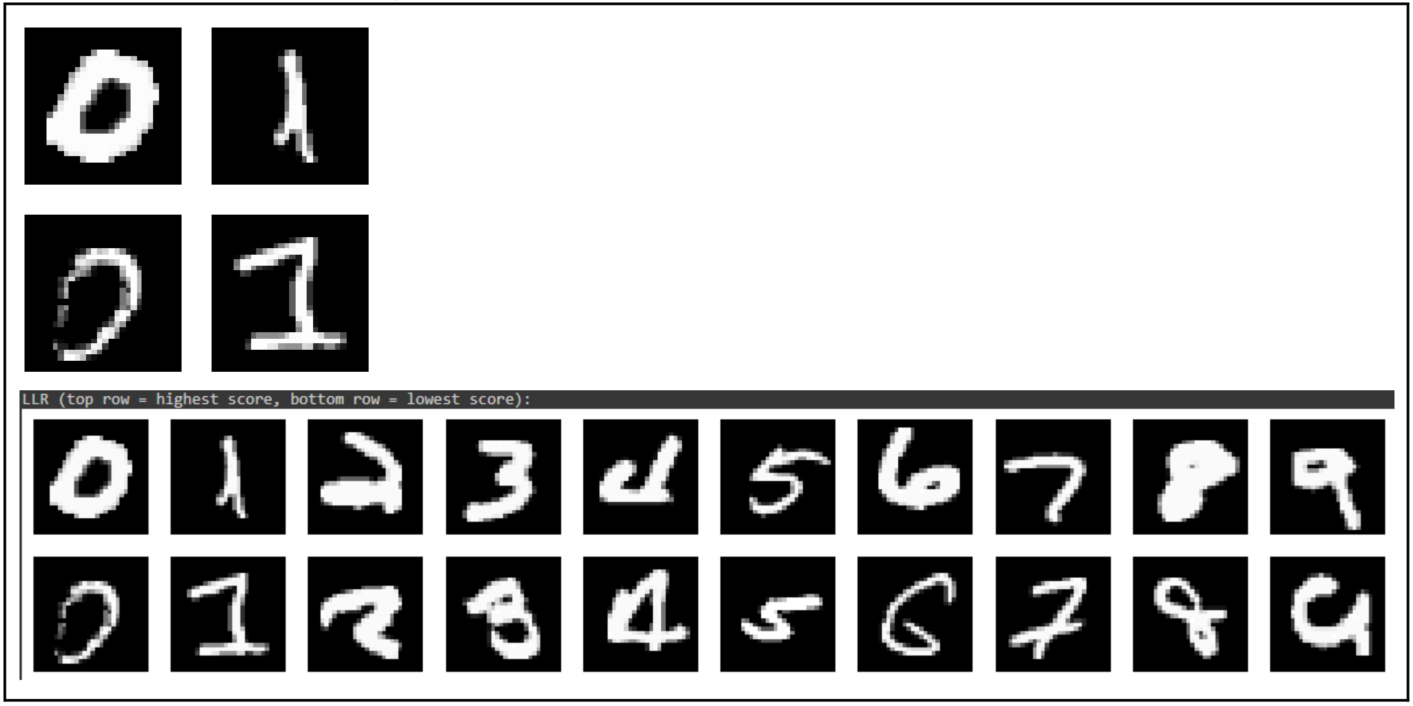
# training samples	LLR	SVM
100[1]	49.8%	53.5%

1,000	18.1%	20.6%
10,000	9.47%	10.8%
60,000	7.41%	8.16%

1. The error at 100 samples are provided for checking your method. May get slightly different results due to not converging.

b. Error visualization

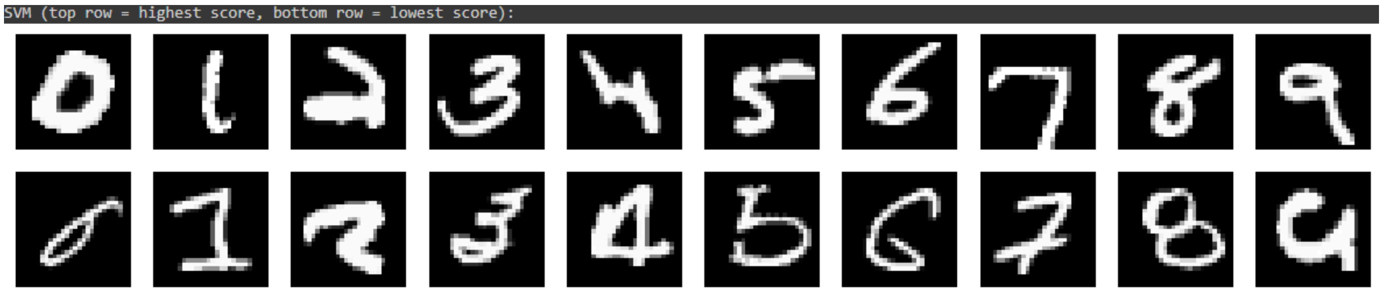
LLR (top row = highest score, bottom row = lowest score)[1]



LLR (top row = highest score, bottom row = lowest score):

1. I’ve displayed what I get for 0 and 1. You should show 0 through 9.

SVM (top row = highest score, bottom row = lowest score)

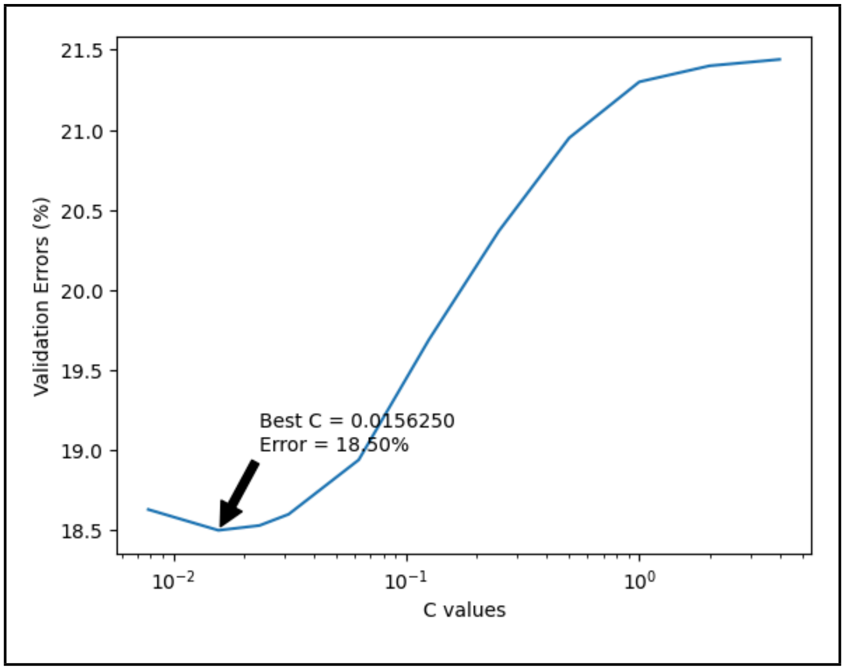


```
SVM (top row = highest score, bottom row = lowest score):
```

c. **Parameter selection experiments**

	SVM
Best C value	0.015625
Validation error (%)	18.50%
Test error (%)	8.28%

Plot C value vs validation error for values tested



3. Temperature Regression

a. Linear regression test

Test RMSE

	Linear regression
Original features[1]	2.0241
Normalized features	2.0365

Why might normalizing features in this way not be as helpful as it is for KNN?

Normalization is crucial for KNN because it ensures all features contribute equally to distance calculations, which matters for KNN. On the other hand, Ridge regression's variance is far less important due to its Regularization: Ridge — scikit-learn 1.5.2 documentation. As a result, normalizing Ridge will have little affect since scaling features does not impact Ridge.

[1] You should get an RMSE in the range of 1.5 to 3 for the original features

b. Feature selection results

Feature Rank	Feature number	City	Day
1	361	Cleveland	-1
2	347	Minneapolis	-1
3	334	Chicago	-1
4	307	Omaha	-2
5	264	Minneapolis	-2
6	241	Albany	-3
7	175	Boston	-3
8	37	Virginia Beach	-5
9	19	Queens	-5
10	9	Boston	-5

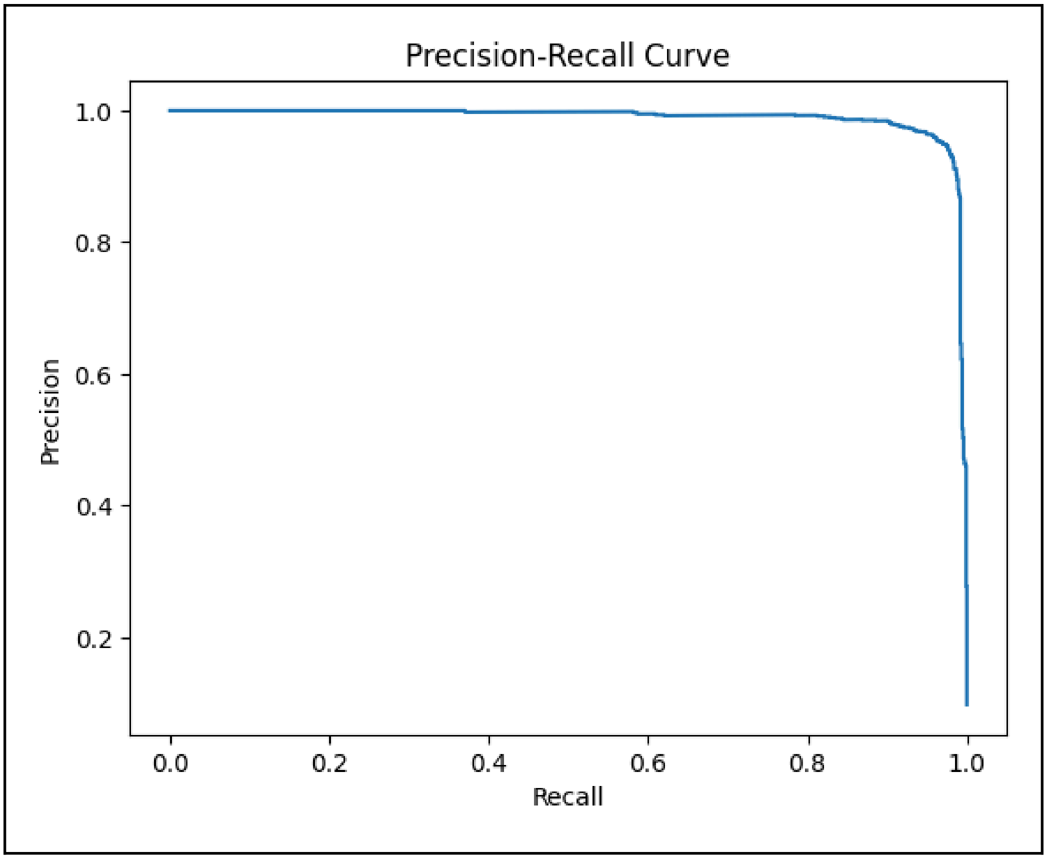
Test error using only the 10 most important features for regression

	Linear Regression
RMS Error	2.1965

4. Stretch Goals

a. PR and ROC curves

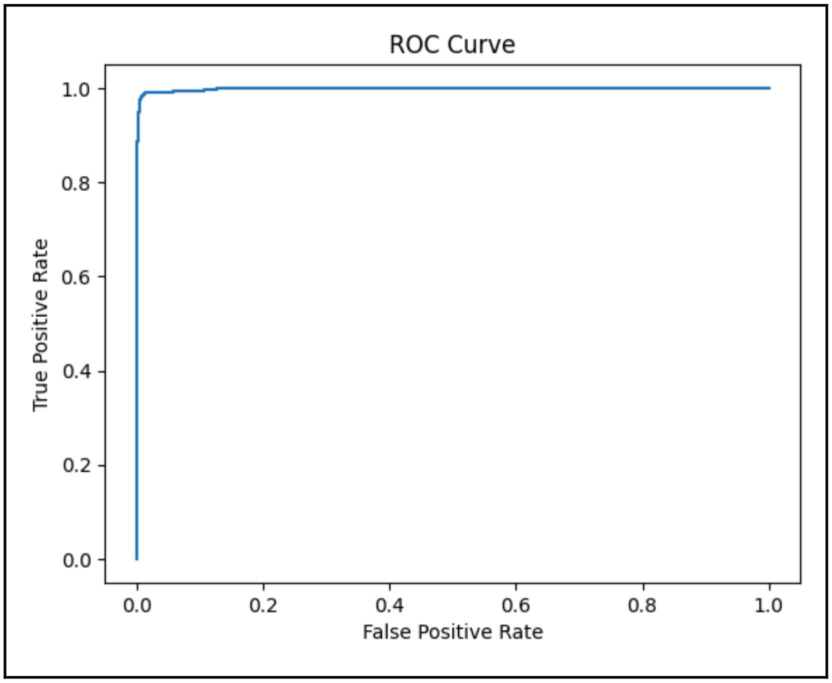
PR plot



Average Precision

0.99

ROC plot

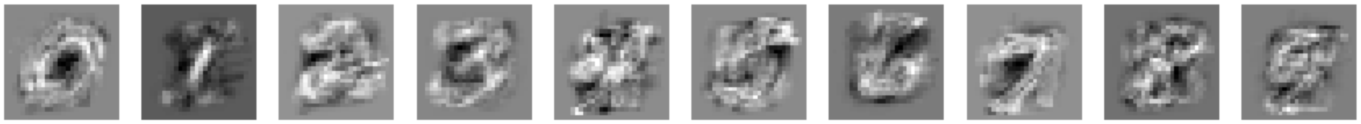


Area under the curve (AUC)

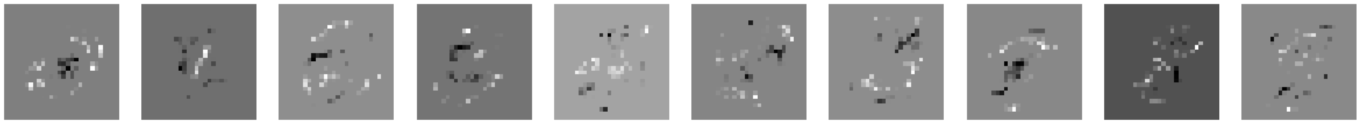
1.00

b. Visualize weights

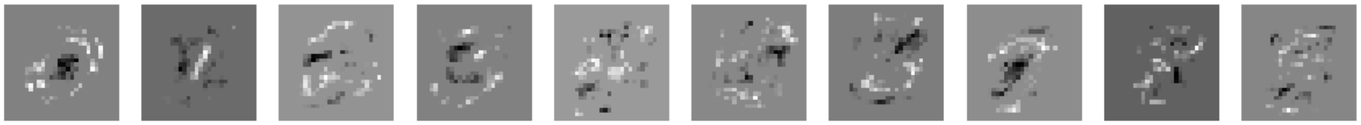
LLR - L2



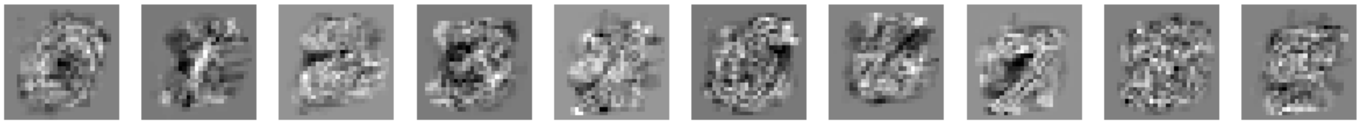
LLR - L1



LLR - elastic



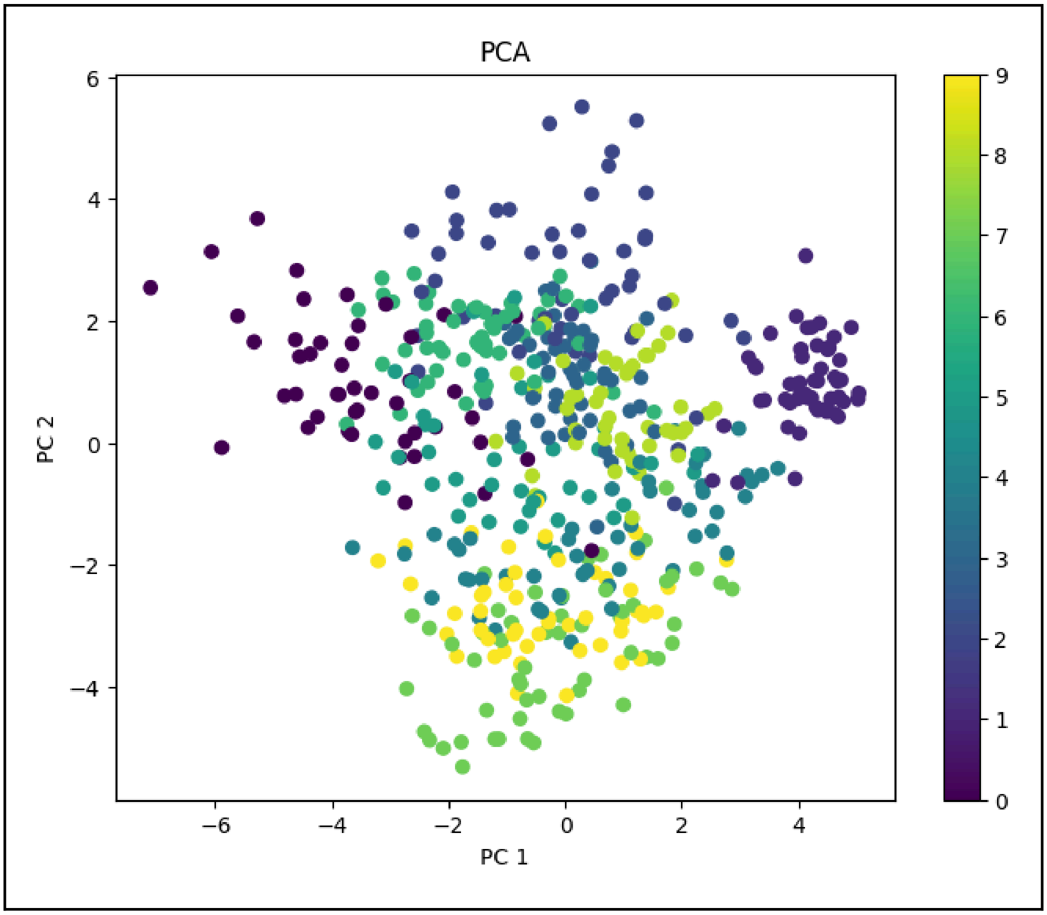
SVM



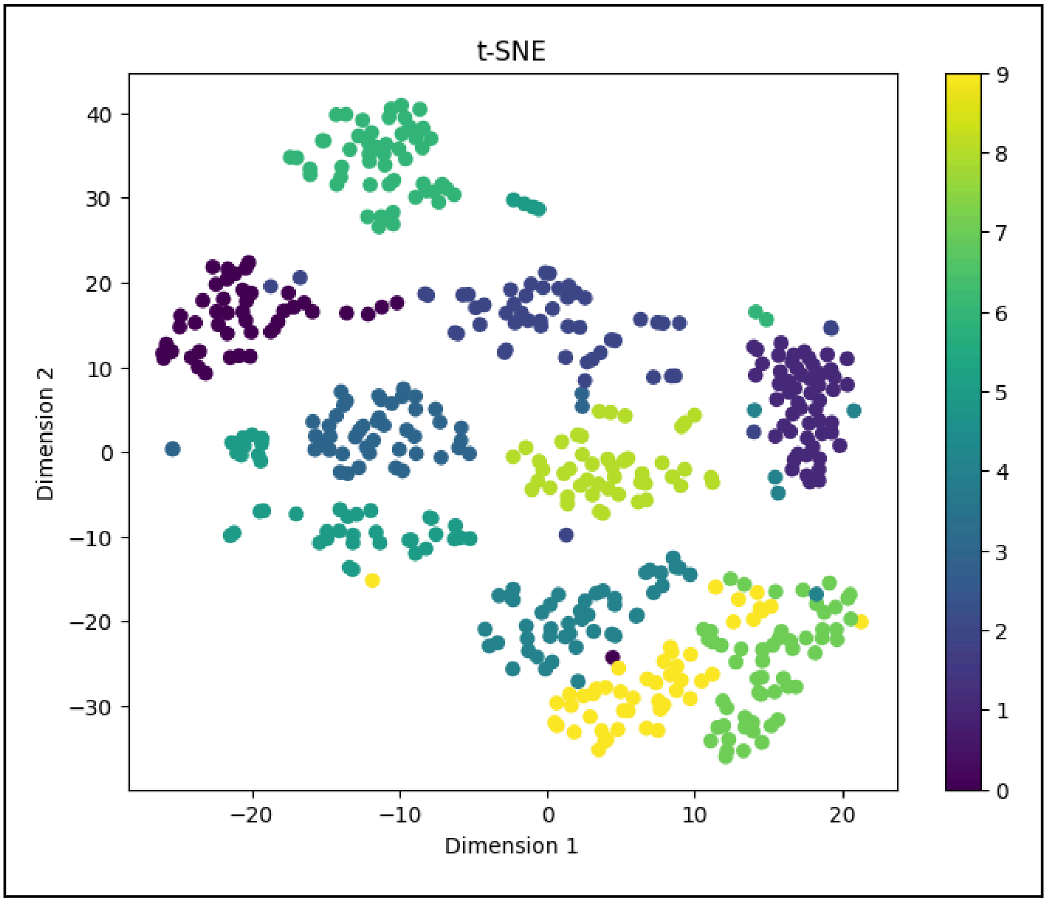
c. Other embeddings

Display 2+ plots for TSNE, MDA, and/or LDA, and copy PCA plot from 1b here.

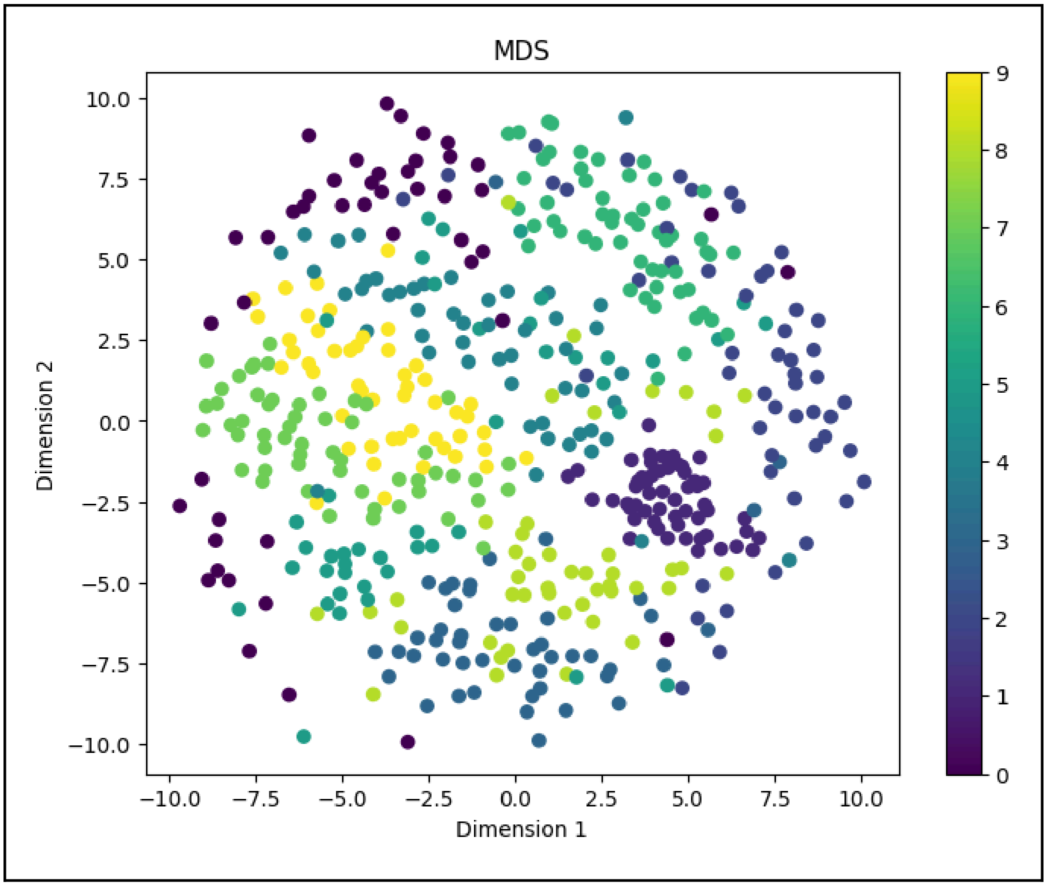
PCA



t-SNE


t-SNE
Dimension 2
Dimension 1
40
30
20
10
0
−10
−20
−30
−20
−10
0
10
20
9
8
7
6
5
4
3
2
1
0

MDS



d. One city is all you need

City

St. Louis

Test error using features only from that city

3.4076

Explain your process (in words):

Select the best city for predicting future temperatures by training a Ridge model on temperature data from the previous five days of each city and evaluate performance using RMSE on a validation set. Once the best city is identified, the model is chosen to train on the best city's data and tested on a separate test set.

e. Compare linear SVM and SVM with RBF kernel

Test accuracy (%)

# training samples	SVM-Linear	SVM-RBF
100	53.54%	49.02
1,000	20.55%	13.09
10,000	10.75%	3.88
60,000	8.16%	2.08

Acknowledgments / Attribution

List any outside sources for code or ideas or “None”.

ChatGPT (Stated in code where)
https://www.geeksforgeeks.org/understanding-the-predictproba-function-in-scikit-learns-svc/
https://www.slingacademy.com/article/numpy-creating-an-array-with-true-false-based-on-an-existing-array/