Gisette Classification Task

GISETTE is a handwritten digit recognition problem.

The problem is to separate the highly confusable digits '4' and '9'.

Presented by: Vipul Chalotra, Eoan O'Dea, and Nataliia Ryvak

Contents

Experimental Design of Data
as explained by the provider of the data

- Normalized the database to get the pixel values in the range [0, 1].
- Threshold values below 0.5 to increase data sparsity.
- Constructed a feature set, which consists of the original variables (normalized pixels) plus a randomly selected subset of products of pairs of variables. The pairs were sampled such that each pair member is normally distributed in a region of the image slightly biased upwards. The rationale behind this choice is that pixels that are discriminative of the “four/nine” separation are more likely to fall in that region.
- Eliminated all features that had only zero values.
- Selected all the original pixels and complemented them with pairs to attain the number of 2500 features.
- Another 2500 pairs were used to construct “probes”: the values of the features were individually permuted across patterns (column randomization). In this way we obtained probes that are similarly distributed to the other features.
- Randomized the order of the features.
- Quantized the data to 1000 levels.

Working Dataset

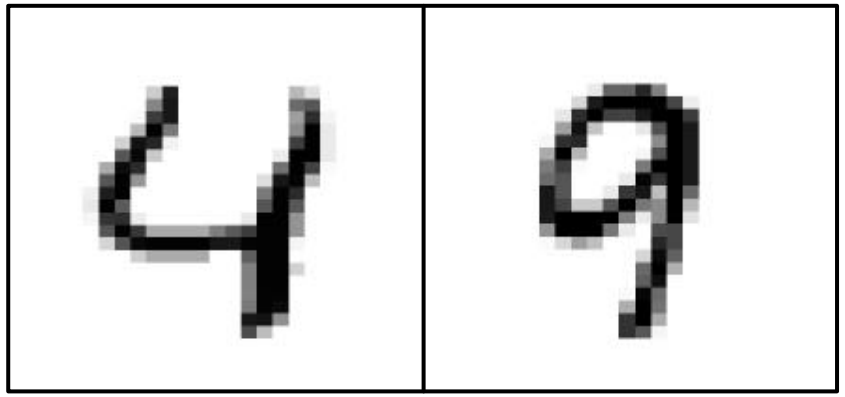


Figure: Two examples of digits from the MNIST database.

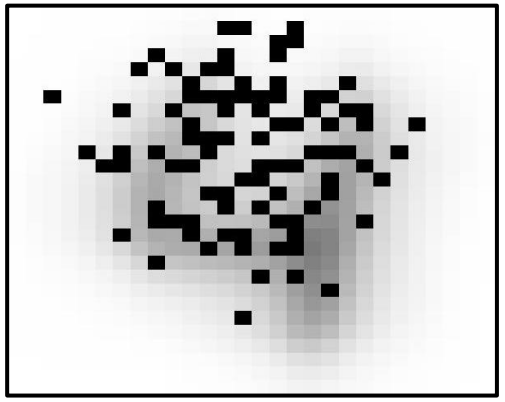


Figure: Example of a randomly selected subset of pixels in the region of interest.

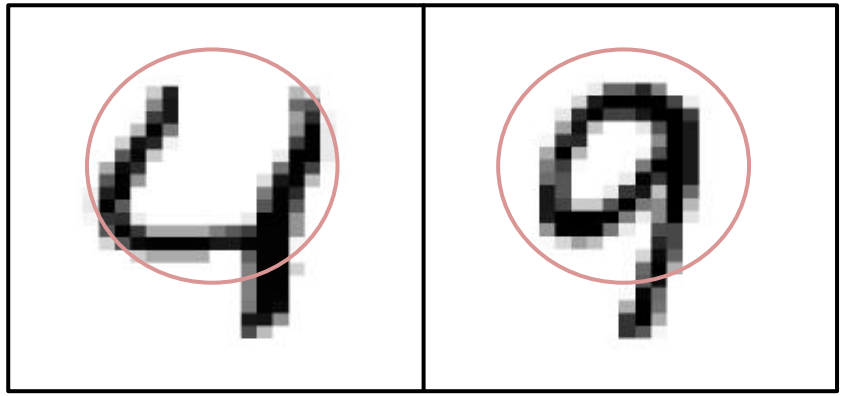
Table: Number of Examples and Class Distribution

	Positive Label	Negative Label	Total
Training Set	3000	3000	6000
Testing Set	500	500	1000
All Data	3500	3500	7000

Table: Input Variables & Variable Statistics

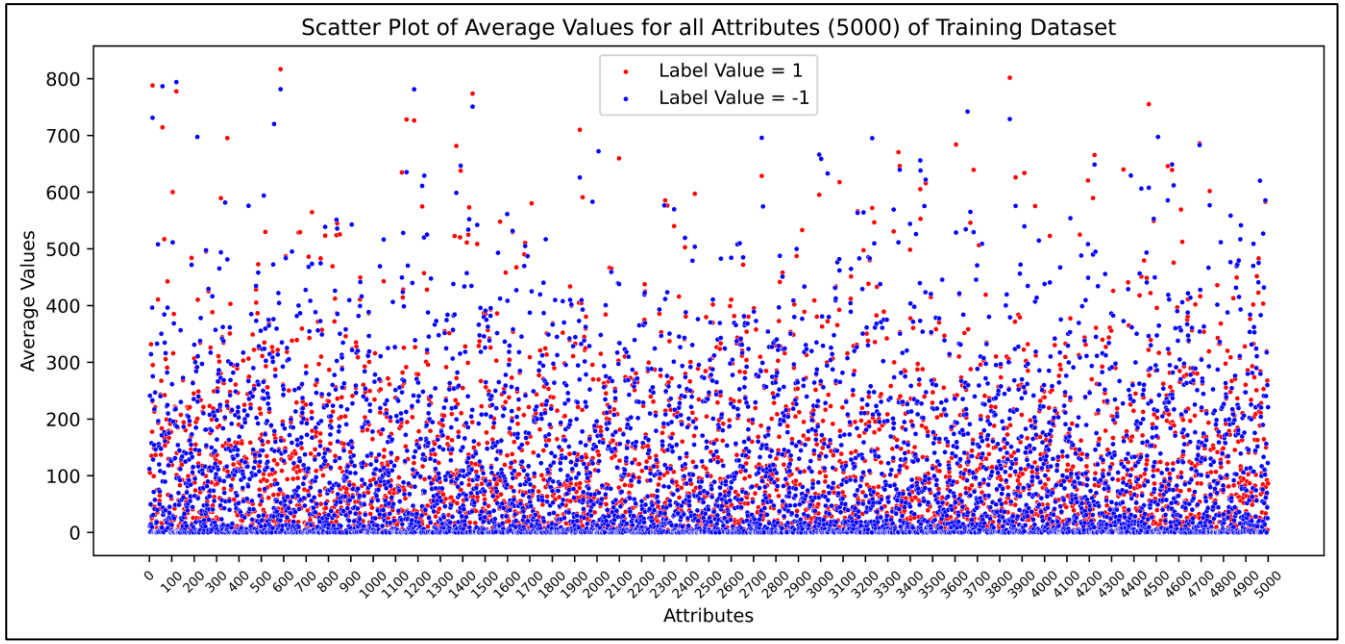
Real Variables	Random Probes	Total
2500	2500	5000

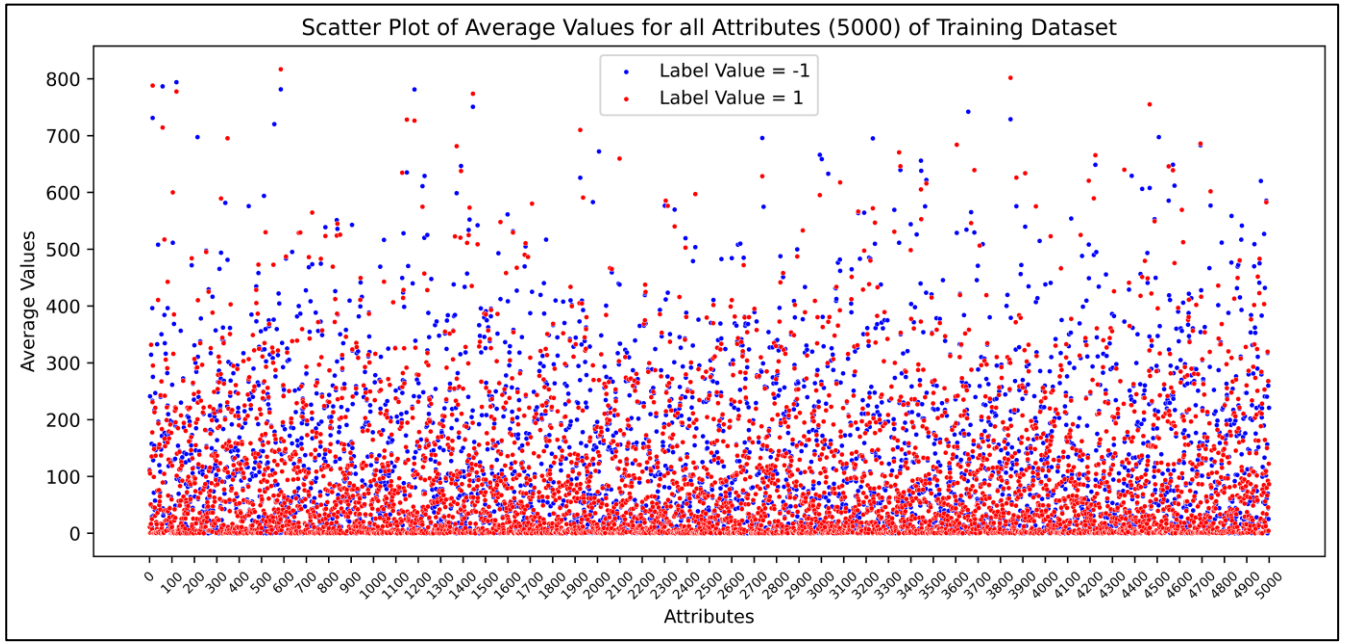
Dimensionality Reduction
using average values across features



- The features refer to pixels and were selected randomly from the area slightly biased upwards.
- The difference between the features will be decisively high near the top center area of the images.

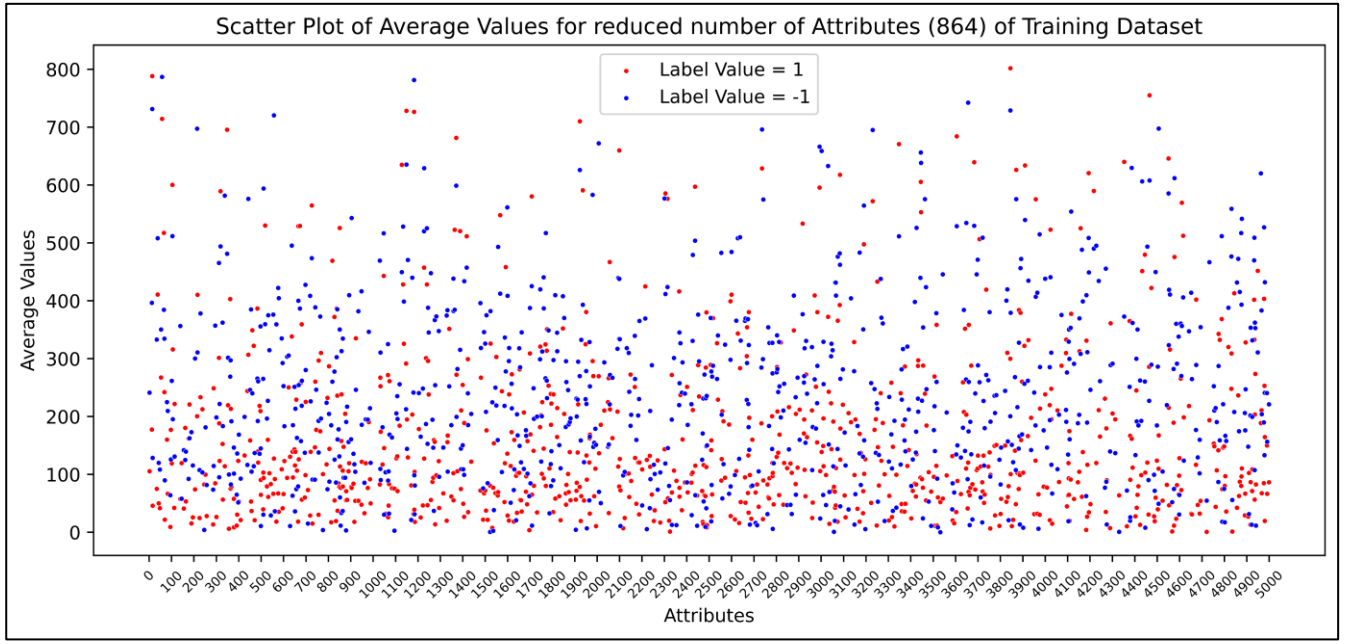
- Separated our testing set into two sets based on the label values: 1 or -1
- Performed exploratory analysis for all 5000 features for each separated set
- Calculated difference between the mean values for the two sets
- Filtered the features using the condition that absolute value of the difference in means should be greater than 50
- Reduced from 5000 features to 864 features

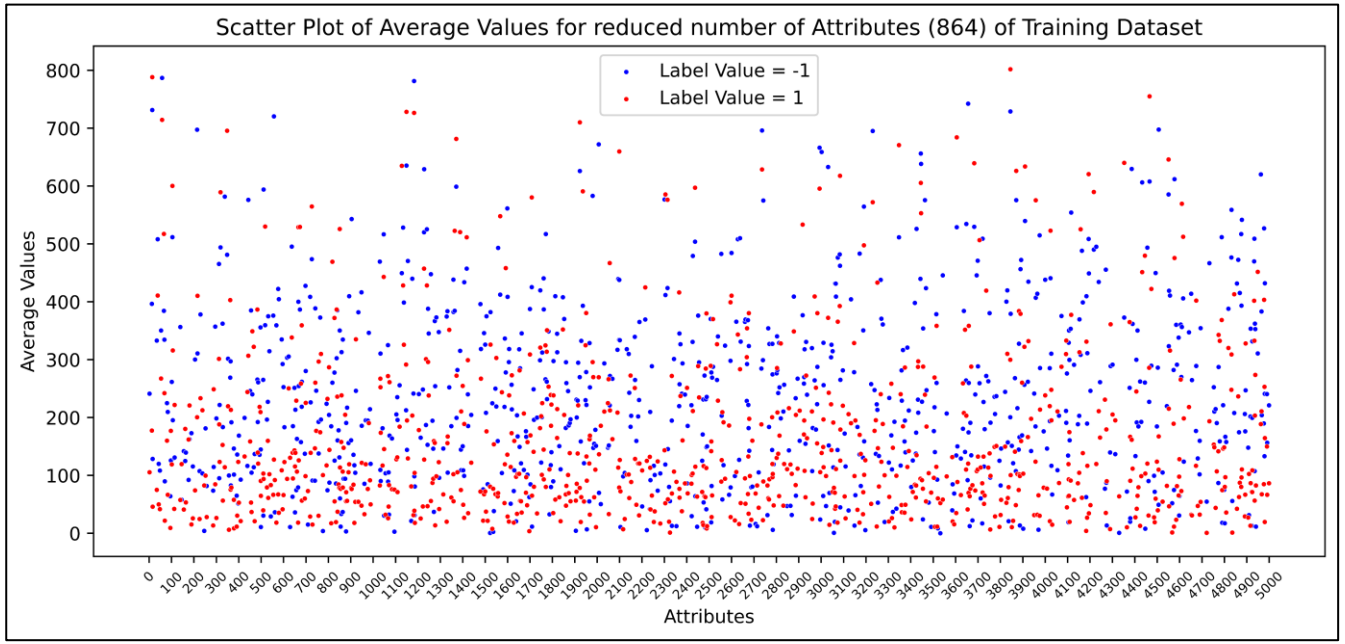




Working with Average Values (Whole Dataset)

- The average values are highly dense near 0 which makes sense for our 87% sparse dataset.
- Nothing else conclusive can be said at this point





Working with Average Values (Reduced Dataset)

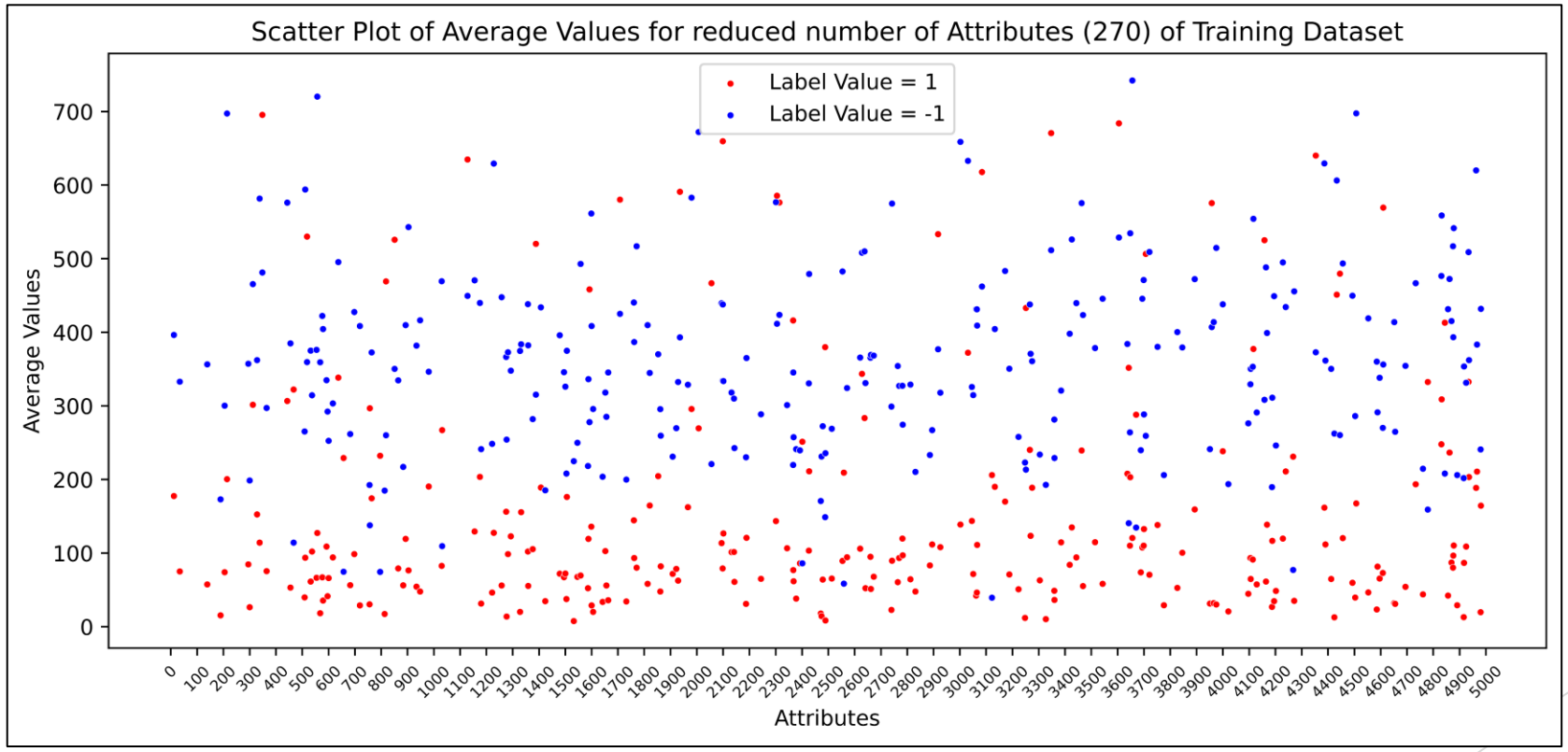
$$|Average| > 50$$

- To some extent, separation is visible.
- Red Points seem to have more average values between 0 & 100.
- Blue Points seem to have more average values above 100.

Working with Average Values (Reduced Dataset)

$$|Average| > 150$$

- Generally, red points lie between average values of 0 & 100 and blue points lie between average values of 200 & 500



Dimensionality Reduction
using Correlation & Multi-Collinearity

- Using correlation of the features to the label, removed features that had less than 15% correlation. Reduced to 542 features.
- Further calculated Variance Inflation Factor (VIF) for 542 features and removed features with VIF greater than 10. Reduced to 299 features.

Table: Top 10 Correlation Values

Feature	Correlation
3656	0.671371
557	0.646177
3975	0.603175
4507	0.577779
511	0.577598
2742	0.569261
3002	0.568450
1228	0.554855
904	0.549525
4271	0.539893

Table: Bottom 10 VIF Values

Feature	VIF	Tolerance
60	1.623279	0.616037
1125	1.691646	0.591140
593	1.716827	0.582470
391	1.761319	0.567756
4835	1.887546	0.529789
1375	1.887829	0.529709
1565	1.904920	0.524956
458	2.024286	0.494001
4197	2.064420	0.484398
2615	2.092110	0.477986

Dimensionality Reduction
using Principal Component Analysis (PCA)

- Performed PCA on our dataset to obtain Principal Components with the following explained variances:

Table: Top 10 Variance Contributions

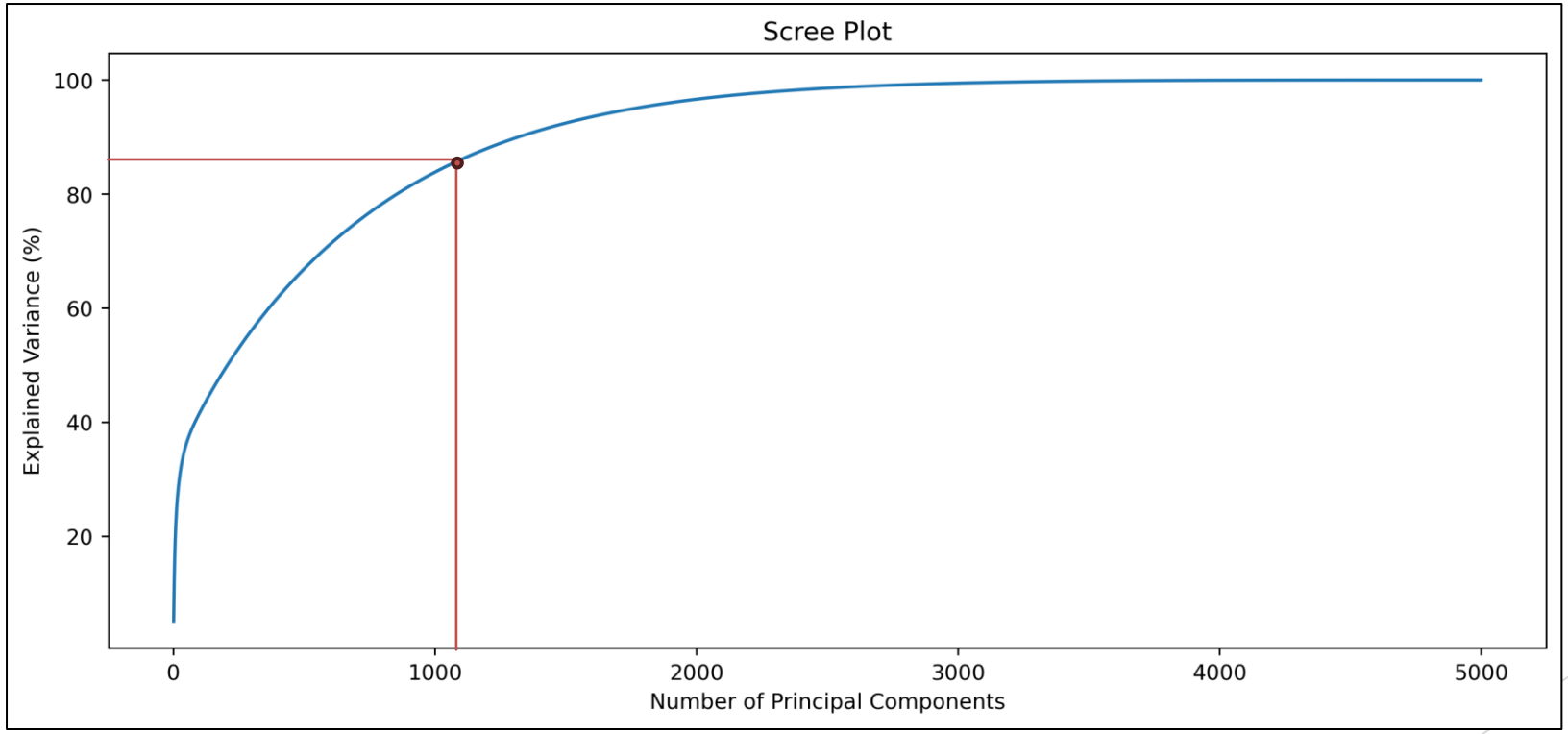
Principal Component	Variance
1	5.111313
2	3.733727
3	3.023898
4	2.326494
5	2.109193
6	1.705387
7	1.479375
8	1.282750
9	1.102118
10	0.955059

Table: Cumulative Variances per 500 Principal Components

Principal Component	Cumulative Variance
1	5.111313
500	66.873663
1000	83.827664
1500	92.441109
2000	96.620063
2500	98.569556
3000	99.453533
3500	99.826127
4000	99.962054
4500	99.997152
5000	100.00

Dimensionality Reduction
using Principal Component Analysis (PCA)

- Using PCA to capture 85% explained Cumulative Variance, reduced 5000 features to 1051 Principal Components.



Towards Machine Learning

- Binary Classification Task.
- Decided to go with Regression models and Support Vector Machines.
- Used scikit-learn Machine Learning library to perform supervised learning.
- After performing Principal Component Analysis, implemented functionalities of the scikit-learn library to create machine learning pipelines.
- Performed classification using Logistic Regression, Linear SVM, and polynomial SVMs of degrees 2, 3, and 4.

Percentage Accuracies & Times

Table: Whole Data Run Outcomes

	Logistic Regression	SVM (Linear)	SVM (Deg 2 Poly)	SVM (Deg 3 Poly)	SVM (Deg 4 Poly)
Accuracy on Training Set	100.0	100.0	99.8	99.783	99.2
Accuracy on Testing Set	97.8	97.6	98.0	96.7	94.1
Fitting Time (in seconds)	8.732	53.156	83.206	110.396	140.337

Table: Mean-Reduced Data Run Outcomes

	Logistic Regression	SVM (Linear)	SVM (Deg 2 Poly)	SVM (Deg 3 Poly)	SVM (Deg 4 Poly)
Accuracy on Training Set	100.0	100.0	99.15	98.216	96.15
Accuracy on Testing Set	97.0	97.0	97.39	95.8	93.89
Fitting Time (in seconds)	3.519	4.744	6.865	9.914	11.208

Percentage Accuracies & Times

Table: Correlation-Reduced Data Run Outcomes

	Logistic Regression	SVM (Linear)	SVM (Deg 2 Poly)	SVM (Deg 3 Poly)	SVM (Deg 4 Poly)
Accuracy on Training Set	100.0	100.0	98.63	97.3	95.717
Accuracy on Testing Set	95.8	95.39	96.8	95.1	93.4
Fitting Time (in seconds)	3.865	2.426	3.426	6.377	7.338

Correlation Calculation Time (in seconds)	342.294

Table: Correlation & Multi-Collinearity-Reduced Data Run Outcomes

	Logistic Regression	SVM (Linear)	SVM (Deg 2 Poly)	SVM (Deg 3 Poly)	SVM (Deg 4 Poly)
Accuracy on Training Set	98.116	NA	98.55	96.816	93.85
Accuracy on Testing Set	95.89	NA	97.39	95.19	90.9
Fitting Time (in seconds)	6.876	>1200	2.516	4.111	5.261

VIF Calculation Time (in seconds)	182.105

Percentage Accuracies & Times

Table: PCA-Reduced Data Run Outcomes

	PCA Calculation Time (in seconds)	87.278

	Logistic Regression	SVM (Linear)	SVM (Deg 2 Poly)	SVM (Deg 3 Poly)	SVM (Deg 4 Poly)
Accuracy on Training Set	100.0	100.0	99.9	99.85	99.53
Accuracy on Testing Set	96.3	96.2	96.89	96.0	80.2
Pipeline Time (in seconds)	92.475	88.809	116.878	127.437	132.562
Fitting Time (in seconds)	~5	~2	~30	~40	~45

- Best Accuracy on Training Set: 100.0% (with 97.8% on Testing Set)
- Best Accuracy on Testing Set: 98.0% (with 99.8% on Training Set)
- Overall Best Model Performance: SVM (Deg 2 Poly)
- Overall Worst Model Performance: SVM (Deg 4 Poly)

Key Takeaways

- As the complexity of the model increased, the fitting and predicting times increased. This increase was highly visible in the whole dataset and moderately visible in the PCA-reduced dataset.
- Correlation and Multi-Collinearity Reduction took long times, but the fitting and predicting times after reduction were the best times across all the reduction methods.
- The model obtained after restricting the absolute value of the mean differences to be at least 50 across all the features performed surprisingly well.
- Correlation and Multi-Collinearity Reduction reduced the dataset by most. It reduced the dataset from 5000 features to 299 features.
- PCA Reduction reduced the dataset by least. It reduced the dataset from 5000 features to 1051 principal components capturing 85% variability of the dataset.

The Winner

- Overall, the best performer was the Support Vector Machine using a 2nd degree polynomial kernel when trained on the dataset reduced using Correlation and Multi-Collinearity.
- The decision was a direct consequence of minimal over-fitting, high accuracy on the testing data, and only about 2.5 seconds of fitting and predicting time.

Table: Correlation & Multi-Collinearity Reduced Data and 2nd Degree Polynomial Kernel SVM Run Results

Percentage Accuracy on Training Set	98.55
Percentage Accuracy on Testing Set	97.39
Correlation Calculation Time (in seconds)	342.294
VIF Calculation Time (in seconds)	182.105
Fitting & Predicting Time (in seconds)	2.516

References

- https://archive.ics.uci.edu/dataset/170/gisette
- Class Notes and Predictive Analytics Slides
- https://towardsdatascience.com/using-principal-component-analysis-pca-for-machine-learning-b6e803f5bf1e
- https://www.youtube.com/watch?v=FgakZw6K1QQ