Student Name: Vlad Dubrovenski, Student ID: 16281273

REPORT Homework 4

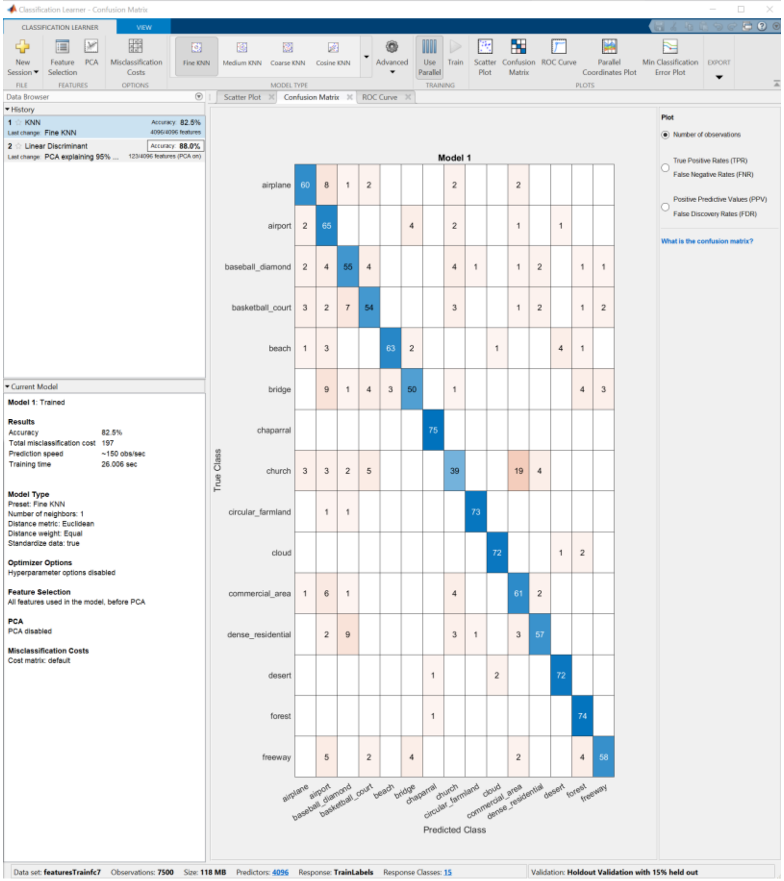
Vlad Dubrovenski

GitHub link to the project: https://github.com/vladi7/ComputerVision

Homework-4: Classification of the remote sensing data set

(1) Use pretrained VGG16 network FC features, find its low dimension embedding of fc1, fc2, and fc3 via PCA+LDA, and compute baseline accuracy with 1-NN classifier, plot the 100x100 affinity map also : (50pts)

In this work, I explored the features of Classification Learner application of Matlab. With this application, I was able to train the classifiers for all the tasks, generate the code for the classifiers, as well as get the characteristics of such models. Below is the User Interface of this application.



1 ☆ KNN Accuracy: 82.5%
Last change: Fine KNN 4096/4096 features

2 ☆ Linear Discriminant Accuracy: **88.0%**
Last change: PCA explaining 95% ... 123/4096 features (PCA on)

Model 1: Trained

Results

Accuracy	82.5%
Total misclassification cost	197
Prediction speed	~150 obs/sec
Training time	26.006 sec

Model Type
Preset: Fine KNN
Number of neighbors: 1
Distance metric: Euclidean
Distance weight: Equal
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Misclassification Costs
Cost matrix: default

Model 1

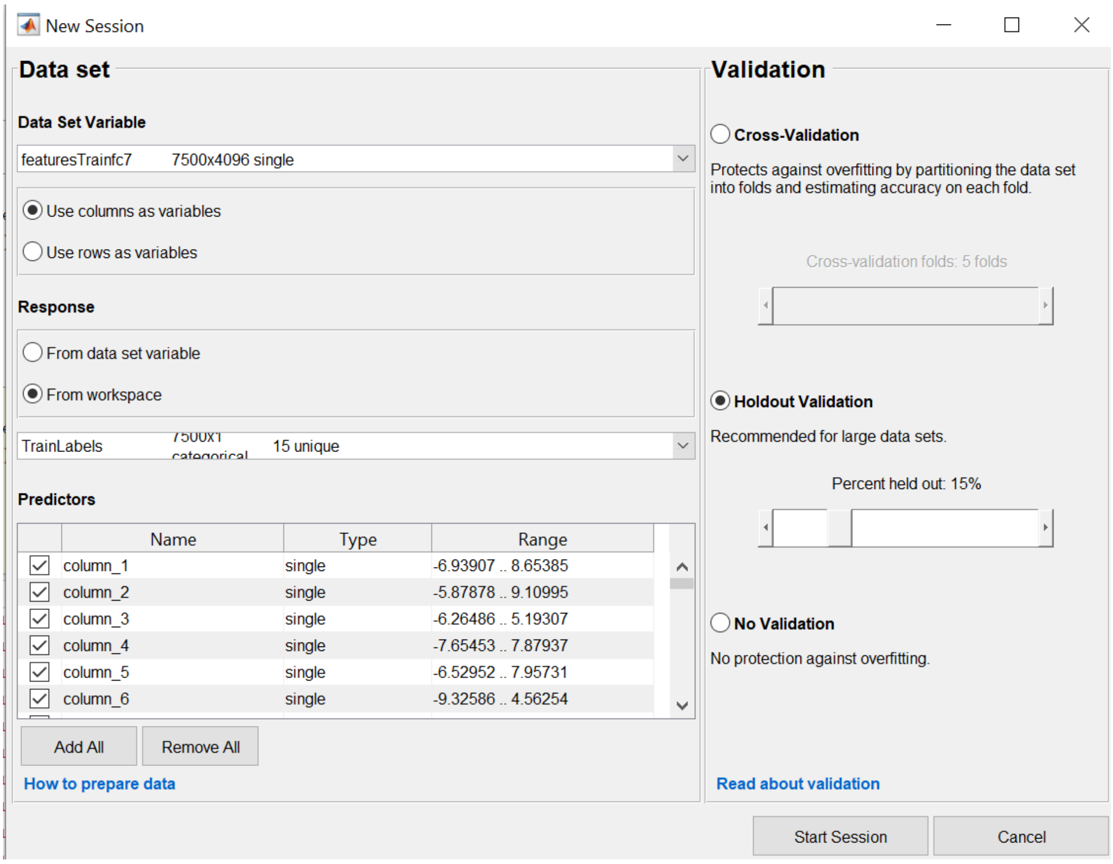
True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	60	8	1	2				2			2				
airport	2	65				4		2			1		1		
baseball_diamond	2	4	55	4				4	1		1	2		1	1
basketball_court	3	2	7	54				3			1	2		1	2
beach	1	3			63	2				1			4	1	
bridge		9	1	4	3	50		1						4	3
chaparral							75								
church	3	3	2	5				39			19	4			
circular_farmland		1	1						73						
cloud										72			1	2	
commercial_area	1	6	1					4			61	2			
dense_residential		2	9					3	1		3	57			
desert							1			2			72		
forest							1							74	
freeway		5		2		4					2			4	58

Plot
● Number of observations
○ True Positive Rates (TPR) False Negative Rates (FNR)
○ Positive Predictive Values (PPV) False Discovery Rates (FDR)

What is the confusion matrix?

Data set: **featuresTrainfc7** Observations: **7500** Size: **118 MB** Predictors: 4096 Response: **TrainLabels** Response Classes: 15 Validation: **Holdout Validation with 15% held out**

As you may notice, I could just use the validation dataset directly from this application with holdout validation. I also was able to include the PCA with this app. That improved my development speed by a lot, however, I still had to modify the code of the resulting classifier to accomplish most of the tasks. Below is how to set the validation set to 15%. I did, of course, split the data in 3 sets as well with code, but I found that using GUI is a lot faster for development and lets me try a lot of possible classifiers.



New Session

Data set

Data Set Variable

featuresTrainfc7 7500x4096 single

● Use columns as variables

○ Use rows as variables

Response

○ From data set variable

● From workspace

TrainLabels 7500x1 categorical 15 unique

Predictors

	Name	Type	Range
☑	column_1	single	-6.93907 .. 8.65385
☑	column_2	single	-5.87878 .. 9.10995
☑	column_3	single	-6.26486 .. 5.19307
☑	column_4	single	-7.65453 .. 7.87937
☑	column_5	single	-6.52952 .. 7.95731
☑	column_6	single	-9.32586 .. 4.56254

Validation

○ **Cross-Validation**

Protects against overfitting by partitioning the data set into folds and estimating accuracy on each fold.

Cross-validation folds: 5 folds

● **Holdout Validation**

Recommended for large data sets.

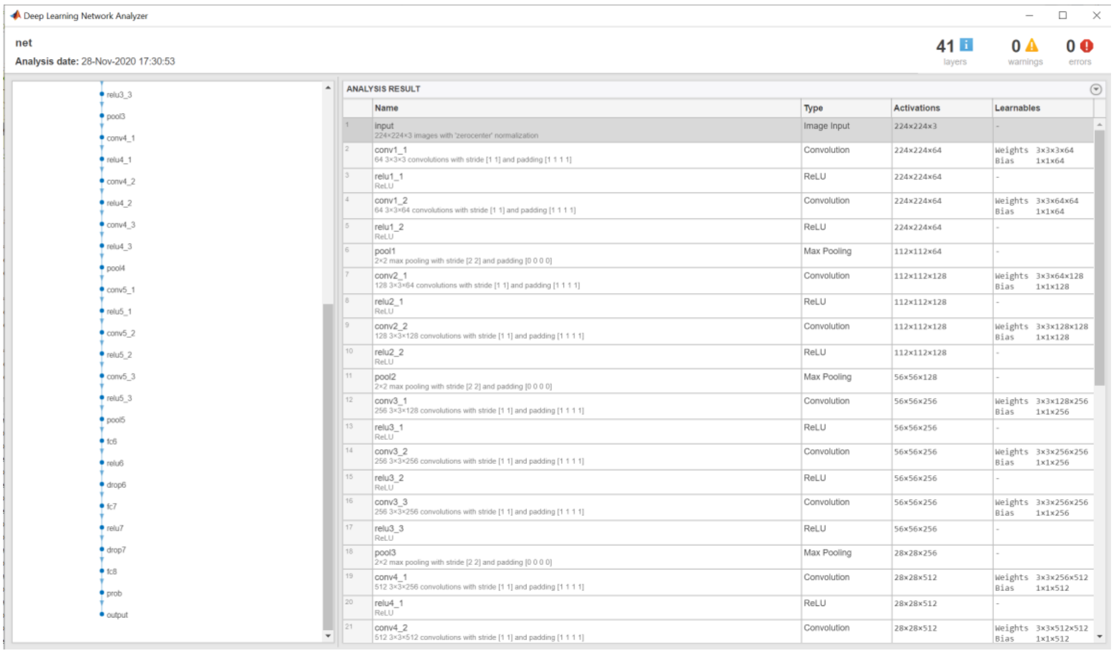
Percent held out: 15%

○ **No Validation**

No protection against overfitting.

To accomplish the following, I used the deep learning network toolbox, VGG16 trained on image net, and standard functions the Matlab provides with version 2020b. Below is the scheme that I obtained with matlab analyzeNetwork() function which shows every layer and its properties.

Student Name: Vlad Dubrovenski,   Student ID: 16281273



Deep Learning Network Analyzer

net

Analysis date: 28-Nov-2020 17:30:53

ANALYSIS RESULT

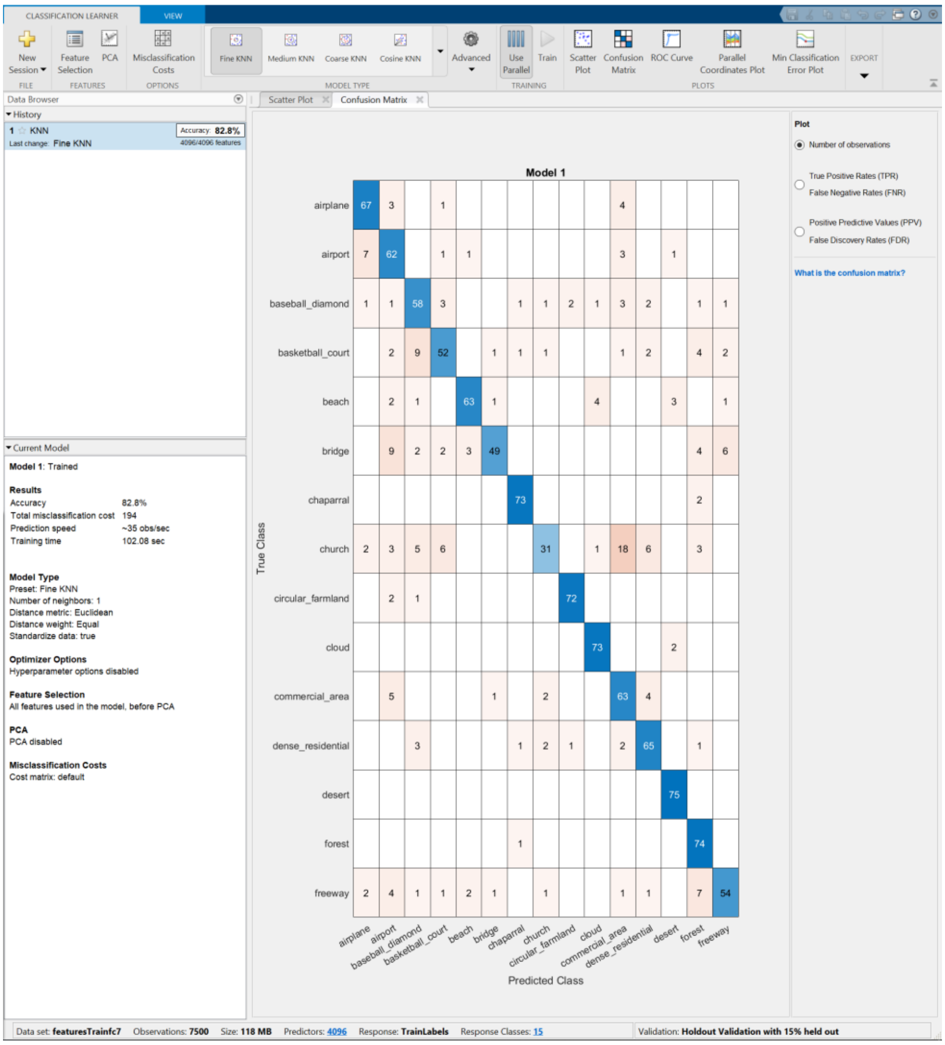
	Name	Type	Activations	Learnables
1	input 224×224×3 images with 'zerocenter' normalization	Image Input	224×224×3	-
2	conv1_1 64 3×3×3 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	224×224×64	Weights 3×3×3×64 Bias 1×1×64
3	relu1_1 ReLU	ReLU	224×224×64	-
4	conv1_2 64 3×3×64 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	224×224×64	Weights 3×3×64×64 Bias 1×1×64
5	relu1_2 ReLU	ReLU	224×224×64	-
6	pool1 2×2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	112×112×64	-
7	conv2_1 128 3×3×64 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	112×112×128	Weights 3×3×64×128 Bias 1×1×128
8	relu2_1 ReLU	ReLU	112×112×128	-
9	conv2_2 128 3×3×128 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	112×112×128	Weights 3×3×128×128 Bias 1×1×128
10	relu2_2 ReLU	ReLU	112×112×128	-
11	pool2 2×2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	56×56×128	-
12	conv3_1 256 3×3×128 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	56×56×256	Weights 3×3×128×256 Bias 1×1×256
13	relu3_1 ReLU	ReLU	56×56×256	-
14	conv3_2 256 3×3×256 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	56×56×256	Weights 3×3×256×256 Bias 1×1×256
15	relu3_2 ReLU	ReLU	56×56×256	-
16	conv3_3 256 3×3×256 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	56×56×256	Weights 3×3×256×256 Bias 1×1×256
17	relu3_3 ReLU	ReLU	56×56×256	-
18	pool3 2×2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	28×28×256	-
19	conv4_1 512 3×3×256 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	28×28×512	Weights 3×3×256×512 Bias 1×1×512
20	relu4_1 ReLU	ReLU	28×28×512	-
21	conv4_2 512 3×3×512 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	28×28×512	Weights 3×3×512×512 Bias 1×1×512

Below are the model features for each FC and both the validation and testing confusion matrices. Also, I of course included the accuracy on testing set.

FC1

KNN Base Line Validation Accuracy + Confusion Matrix + other information

Student Name: Vlad Dubrovenski, Student ID: 16281273



1 ☆ KNN Accuracy: 82.8%
Last change: Fine KNN 4096/4096 features

Current Model

Model 1: Trained

Results

Accuracy	82.8%
Total misclassification cost	194
Prediction speed	~35 obs/sec
Training time	102.08 sec

Model Type
Preset: Fine KNN
Number of neighbors: 1
Distance metric: Euclidean
Distance weight: Equal
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Misclassification Costs
Cost matrix: default

Model 1

Predicted Class

True Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	67	3		1							4				
airport	7	62		1	1						3		1		
baseball_diamond	1	1	58	3			1	1	2	1	3	2		1	1
basketball_court		2	9	52		1	1	1			1	2		4	2
beach		2	1		63	1				4			3		1
bridge		9	2	2	3	49								4	6
chaparral							73							2	
church	2	3	5	6				31		1	18	6		3	
circular_farmland		2	1						72						
cloud										73			2		
commercial_area		5				1		2			63	4			
dense_residential			3				1	2	1		2	65		1	
desert													75		
forest							1							74	
freeway	2	4	1	1	2	1		1			1	1		7	54

Plot
● Number of observations
○ True Positive Rates (TPR)
False Negative Rates (FNR)
○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)

Data set: **featuresTrainfc7** Observations: **7500** Size: **118 MB** Predictors: 4096 Response: **TrainLabels** Response Classes: 15 Validation: **Holdout Validation with 15% held out**

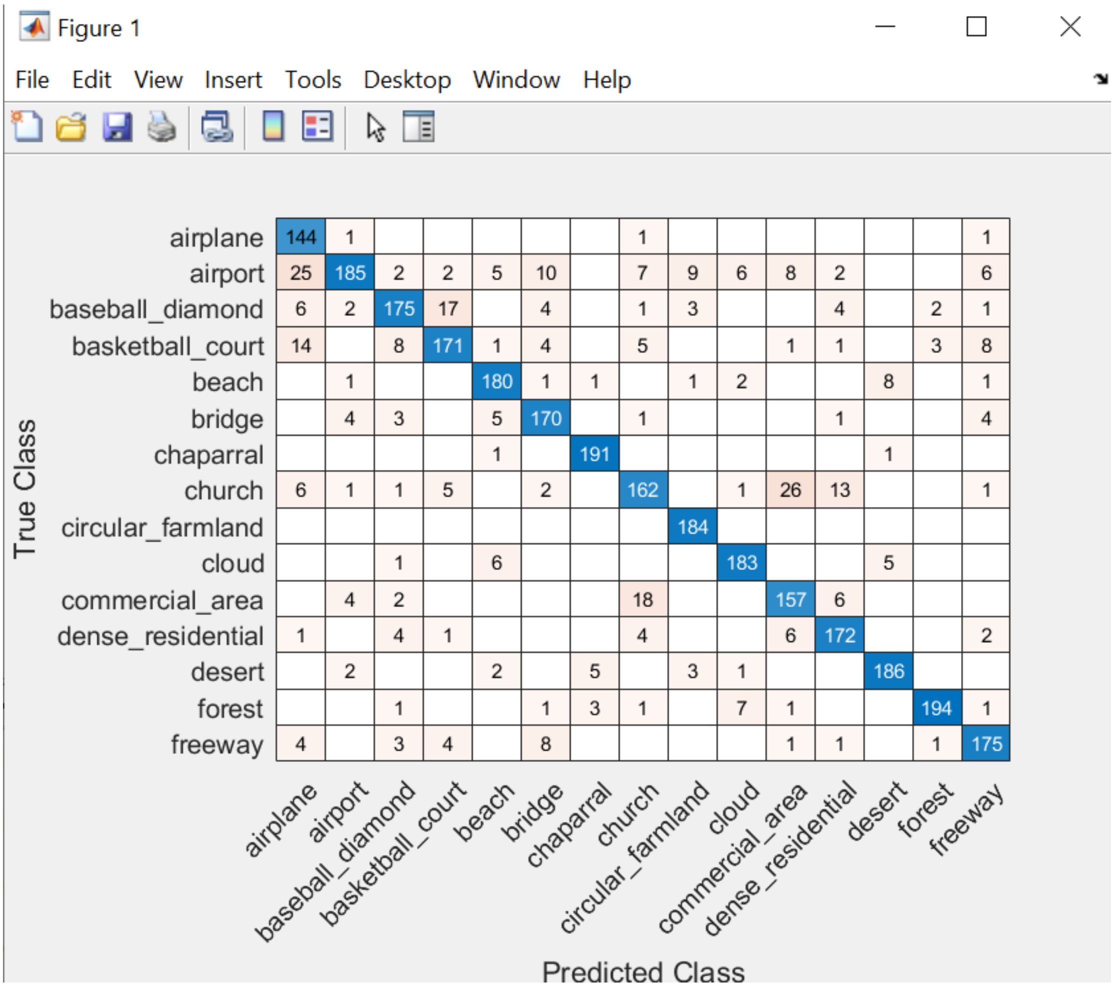
KNN Base Line Testing:

Accuracy:

```
0.8307
```

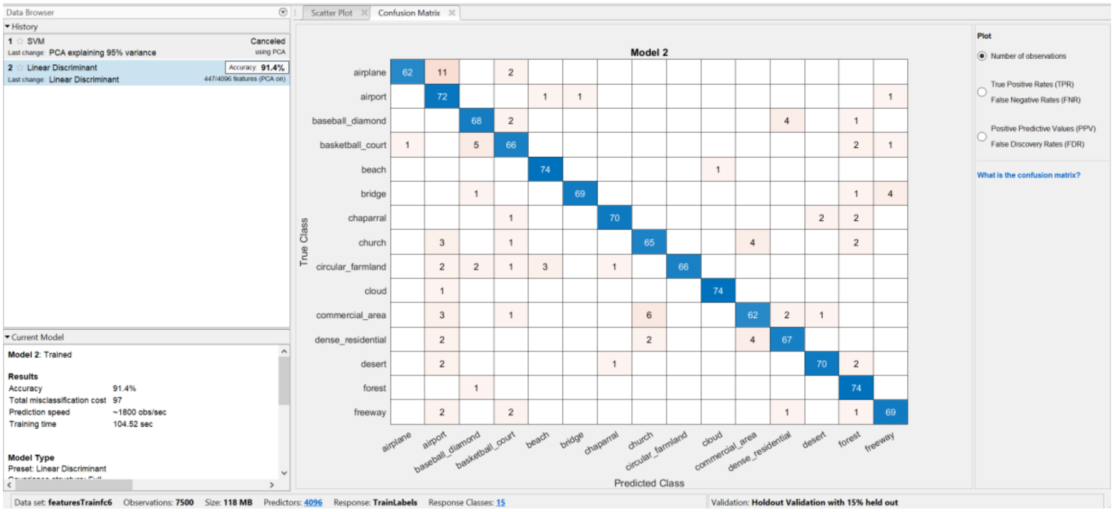
Affinity Map:

Student Name: Vlad Dubrovenski, Student ID: 16281273



True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	144	1						1							1
airport	25	185	2	2	5	10		7	9	6	8	2			6
baseball_diamond	6	2	175	17		4		1	3			4		2	1
basketball_court	14		8	171	1	4		5			1	1		3	8
beach		1			180	1	1		1	2			8		1
bridge		4	3		5	170		1				1			4
chaparral					1		191						1		
church	6	1	1	5		2		162		1	26	13			1
circular_farmland									184						
cloud			1		6					183			5		
commercial_area		4	2					18			157	6			
dense_residential	1		4	1				4			6	172			2
desert		2			2		5		3	1			186		
forest			1			1	3	1		7	1			194	1
freeway	4		3	4		8					1	1		1	175

LDA+PCA Validation Accuracy + Confusion Matrix + other information


Data Browser
History
Scatter Plot
Confusion Matrix
Model 2
True Class
Predicted Class
Plot
○ Number of observations
○ True Positive Rates (TPR)
False Negative Rates (FNR)
○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)
What is the confusion matrix?
Current Model

1 ☆ SVM Canceled
Last change: PCA explaining 95% variance using PCA

2 ☆ Linear Discriminant Accuracy: 91.4%
Last change: Linear Discriminant 447/4096 features (PCA on)

Model 2: Trained

Results

Accuracy	91.4%
Total misclassification cost	97
Prediction speed	~1800 obs/sec
Training time	104.52 sec

Model Type
Preset: Linear Discriminant

True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	62	11		2											
airport		72			1	1									1
baseball_diamond			68	2								4		1	
basketball_court	1		5	66										2	1
beach					74					1					
bridge			1			69								1	4
chaparral				1			70						2	2	
church		3		1				65			4			2	
circular_farmland		2	2	1	3		1		66						
cloud		1								74					
commercial_area		3		1				6			62	2	1		
dense_residential		2						2			4	67			
desert		2					1						70	2	
forest			1											74	
freeway		2		2								1		1	69

Data set: **featuresTrainfc6** Observations: **7500** Size: **118 MB** Predictors: 4096 Response: **TrainLabels** Response Classes: 15 Validation: **Holdout Validation with 15% held out**

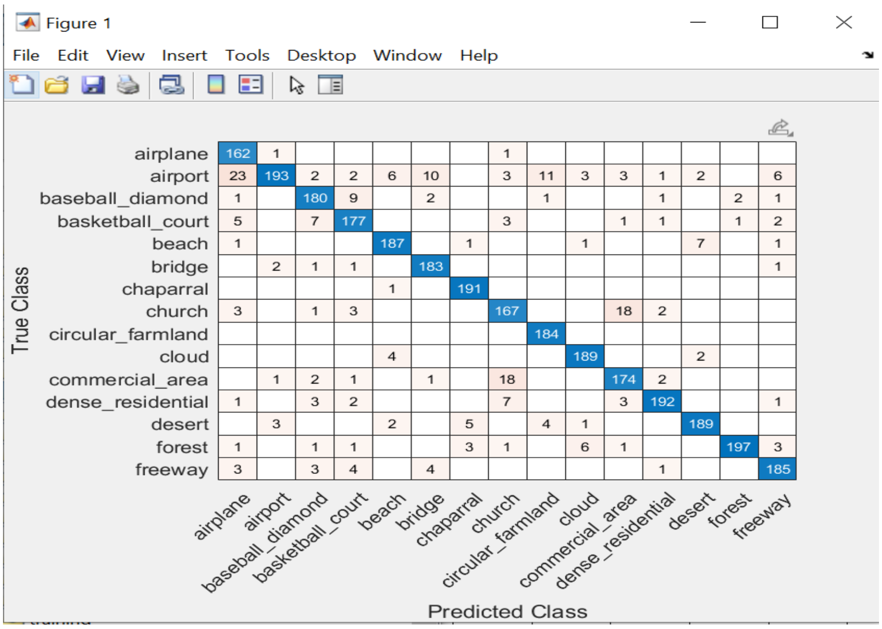
Student Name: Vlad Dubrovenski, Student ID: 16281273

LDA+PCA Testing

Accuracy:

```
0.9167
```

Affinity Map:

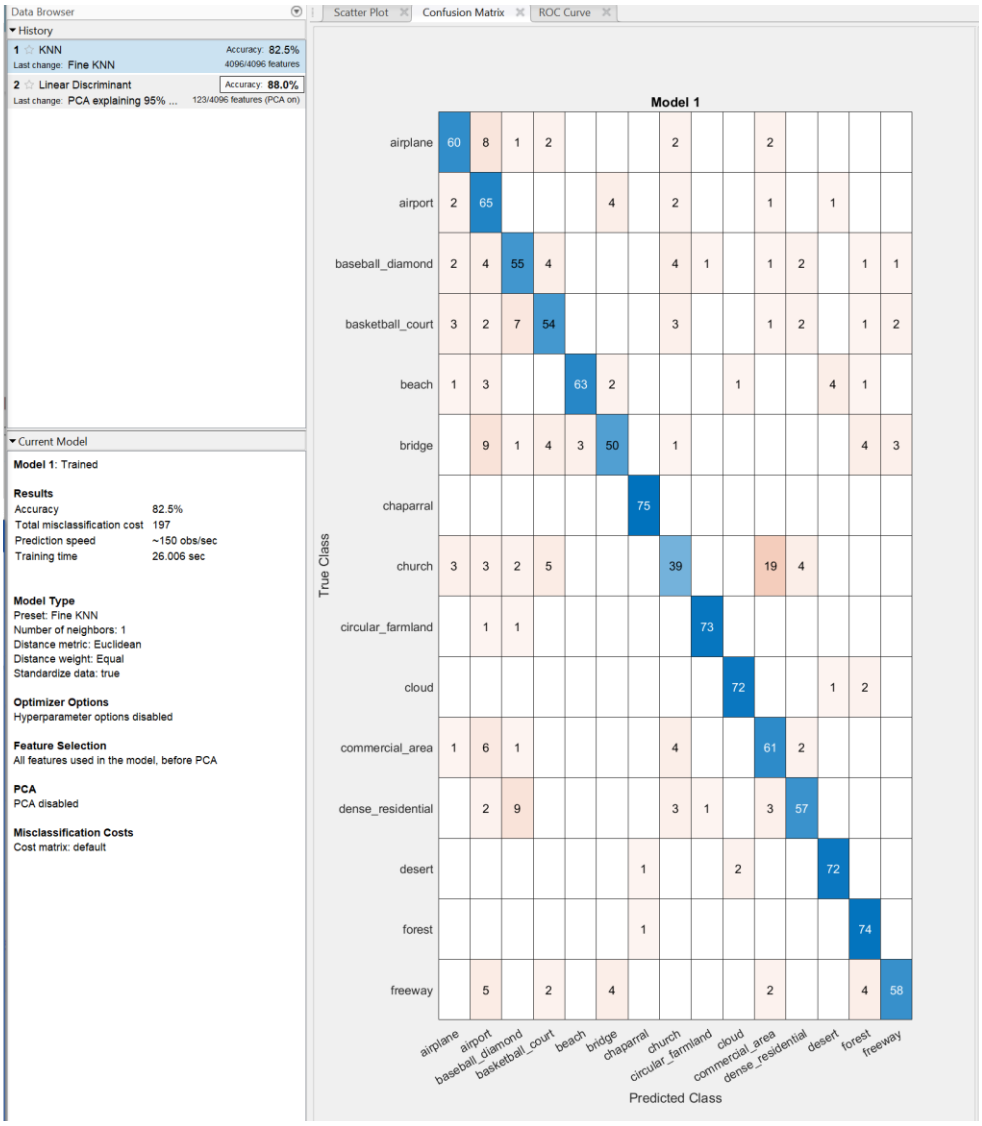


True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	162	1						1							
airport	23	193	2	2	6	10		3	11	3	3	1	2		6
baseball_diamond	1		180	9		2			1			1		2	1
basketball_court	5		7	177				3			1	1		1	2
beach	1				187		1			1			7		1
bridge		2	1	1		183									1
chaparral					1		191								
church	3		1	3				167			18	2			
circular_farmland									184						
cloud					4					189			2		
commercial_area		1	2	1		1		18			174	2			
dense_residential	1		3	2				7			3	192			1
desert		3			2		5		4	1			189		
forest	1		1	1			3	1		6	1			197	3
freeway	3		3	4		4						1			185

FC2

KNN Base Line Validation Accuracy + Confusion Matrix + other information

Student Name: Vlad Dubrovenski, Student ID: 16281273



1 ☆ KNN Accuracy: 82.5%
Last change: Fine KNN 4096/4096 features

2 ☆ Linear Discriminant Accuracy: **88.0%**
Last change: PCA explaining 95% ... 123/4096 features (PCA on)

Current Model

Model 1: Trained

Results

Accuracy	82.5%
Total misclassification cost	197
Prediction speed	~150 obs/sec
Training time	26.006 sec

Model Type
Preset: Fine KNN
Number of neighbors: 1
Distance metric: Euclidean
Distance weight: Equal
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Misclassification Costs
Cost matrix: default

Model 1

True Class / Predicted Class

	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	60	8	1	2				2			2				
airport	2	65				4		2			1		1		
baseball_diamond	2	4	55	4				4	1		1	2		1	1
basketball_court	3	2	7	54				3			1	2		1	2
beach	1	3			63	2				1			4	1	
bridge		9	1	4	3	50		1						4	3
chaparral							75								
church	3	3	2	5				39			19	4			
circular_farmland		1	1						73						
cloud										72			1	2	
commercial_area	1	6	1					4			61	2			
dense_residential		2	9					3	1		3	57			
desert							1			2			72		
forest							1							74	
freeway		5		2		4					2			4	58

:

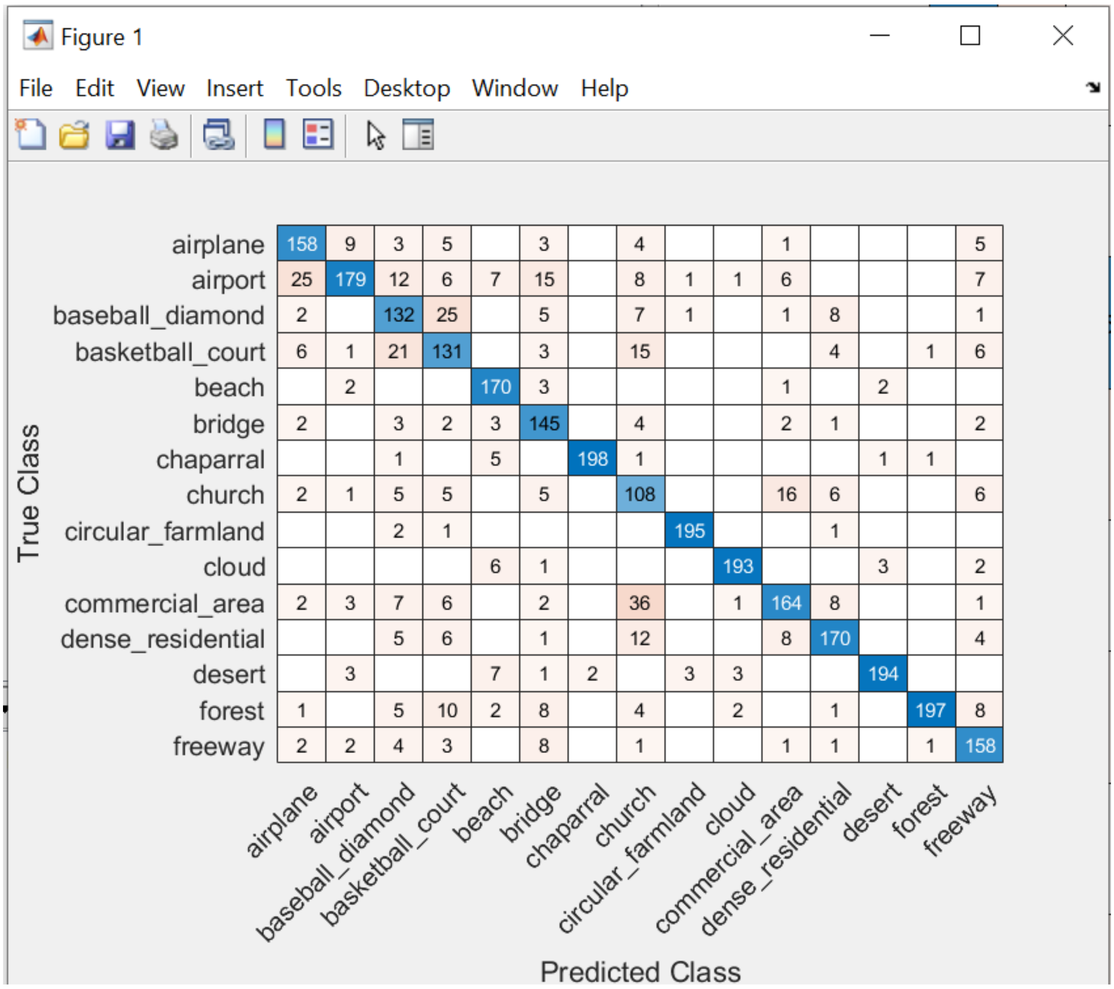
KNN Base Line Testing:

Accuracy:

```
0.8307
```

Affinity Map:

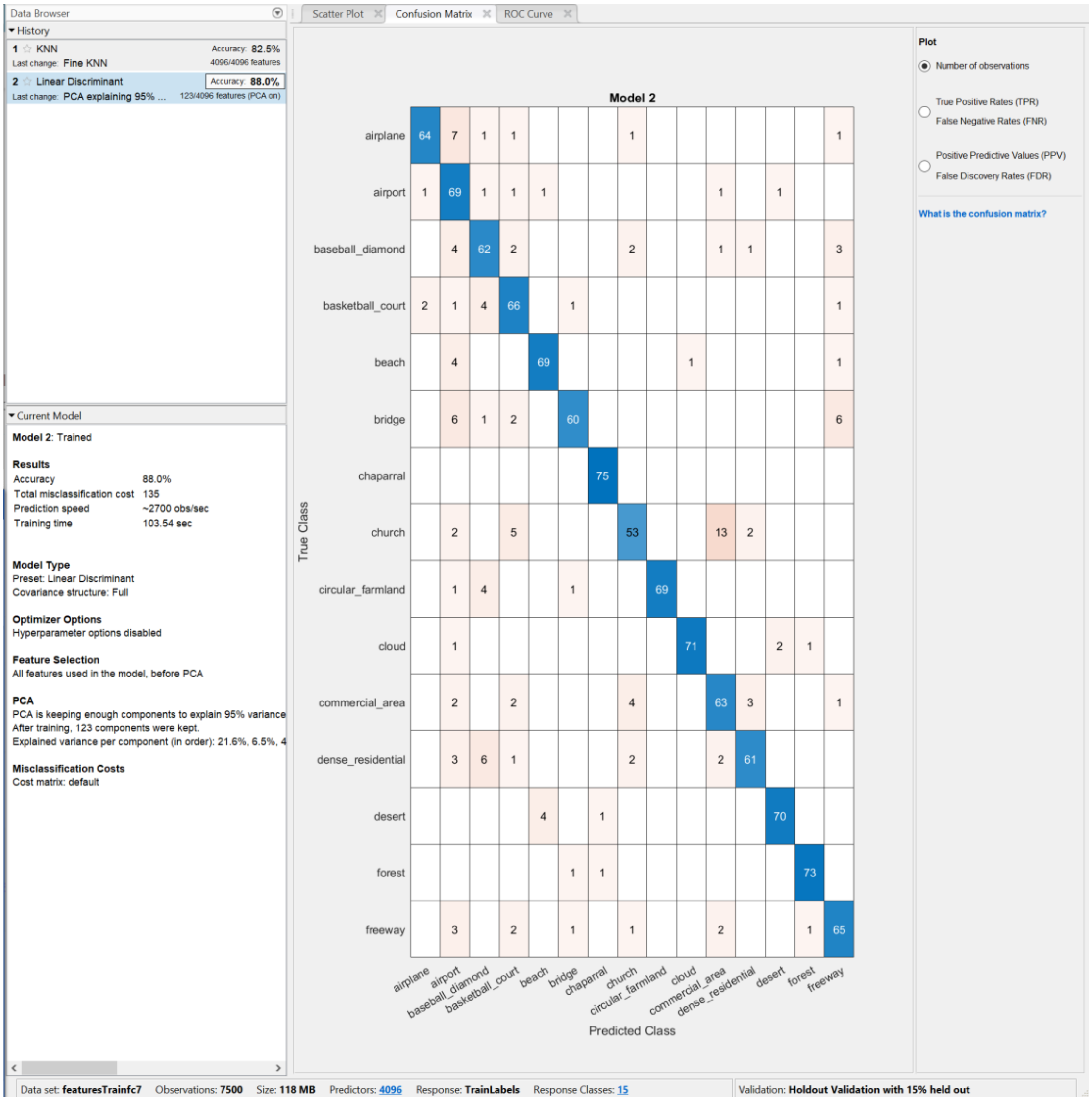
Student Name: Vlad Dubrovenski, Student ID: 16281273



	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	158	9	3	5		3		4			1				5
airport	25	179	12	6	7	15		8	1	1	6				7
baseball_diamond	2		132	25		5		7	1		1	8			1
basketball_court	6	1	21	131		3		15				4		1	6
beach		2			170	3					1		2		
bridge	2		3	2	3	145		4			2	1			2
chaparral			1		5		198	1					1	1	
church	2	1	5	5		5		108			16	6			6
circular_farmland			2	1					195			1			
cloud					6	1				193			3		2
commercial_area	2	3	7	6		2		36		1	164	8			1
dense_residential			5	6		1		12			8	170			4
desert		3			7	1	2		3	3			194		
forest	1		5	10	2	8		4		2		1		197	8
freeway	2	2	4	3		8		1			1	1		1	158

LDA+PCA Validation Accuracy + Confusion Matrix + other information

Student Name: Vlad Dubrovenski, Student ID: 16281273



History

1 ☆ KNN — Accuracy: 82.5%
Last change: Fine KNN — 4096/4096 features

2 ☆ Linear Discriminant — Accuracy: 88.0%
Last change: PCA explaining 95% ... — 123/4096 features (PCA on)

Current Model

Model 2: Trained

Results

Accuracy	88.0%
Total misclassification cost	135
Prediction speed	~2700 obs/sec
Training time	103.54 sec

Model Type
Preset: Linear Discriminant
Covariance structure: Full

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA is keeping enough components to explain 95% variance
After training, 123 components were kept.
Explained variance per component (in order): 21.6%, 6.5%, 4

Misclassification Costs
Cost matrix: default

Model 2

True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	64	7	1	1				1							1
airport	1	69	1	1	1						1		1		
baseball_diamond		4	62	2				2			1	1			3
basketball_court	2	1	4	66		1									1
beach		4			69					1					1
bridge		6	1	2		60									6
chaparral							75								
church		2		5				53			13	2			
circular_farmland		1	4			1			69						
cloud		1								71			2	1	
commercial_area		2		2				4			63	3			1
dense_residential		3	6	1				2			2	61			
desert					4		1						70		
forest						1	1							73	
freeway		3		2		1		1			2			1	65

● Number of observations
○ True Positive Rates (TPR)
False Negative Rates (FNR)
○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)

Data set: **featuresTrainfc7** Observations: **7500** Size: **118 MB** Predictors: **4096** Response: **TrainLabels** Response Classes: **15** Validation: **Holdout Validation with 15% held out**

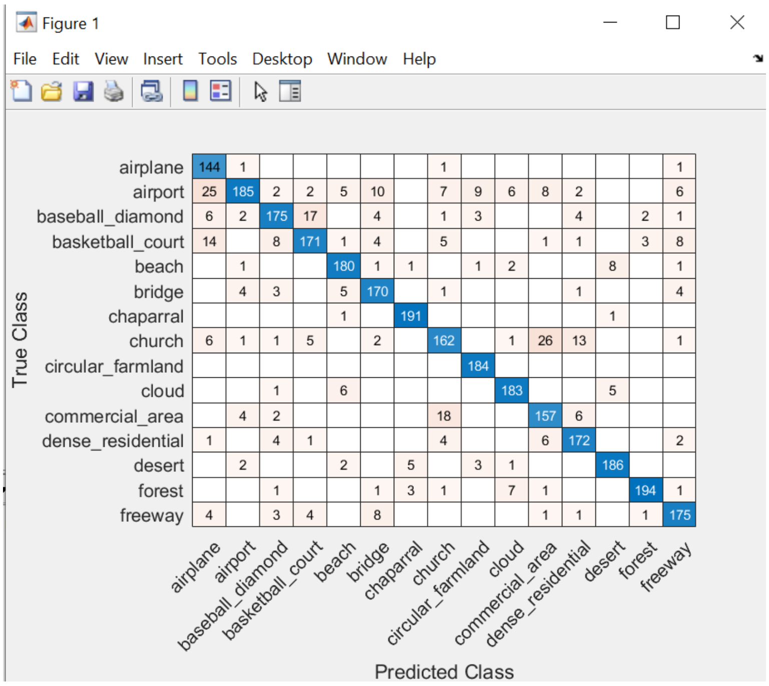
LDA+PCA Testing

Accuracy:

```
0.8763
```

Affinity Map:

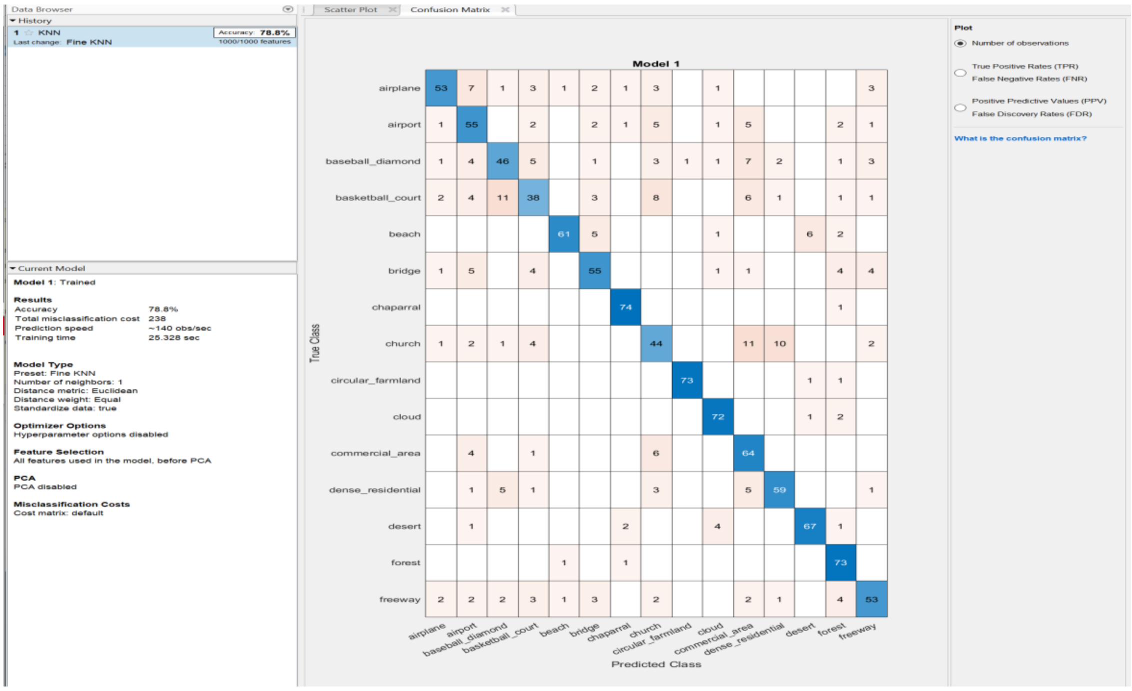
Student Name: Vlad Dubrovenski, Student ID: 16281273



True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	144	1						1							1
airport	25	185	2	2	5	10		7	9	6	8	2			6
baseball_diamond	6	2	175	17		4		1	3			4		2	1
basketball_court	14		8	171	1	4		5			1	1		3	8
beach		1			180	1	1		1	2			8		1
bridge		4	3		5	170		1				1			4
chaparral					1		191						1		
church	6	1	1	5		2		162		1	26	13			1
circular_farmland									184						
cloud			1		6					183			5		
commercial_area		4	2					18			157	6			
dense_residential	1		4	1				4			6	172			2
desert		2			2		5		3	1			186		
forest			1			1	3	1		7	1			194	1
freeway	4		3	4		8					1	1		1	175

FC3

KNN Base Line Validation Accuracy + Confusion Matrix + other information



Model 1: Trained

Results

Accuracy	78.8%
Total misclassification cost	238
Prediction speed	~140 obs/sec
Training time	25.328 sec

Model Type
Preset: Fine KNN
Number of neighbors: 1
Distance metric: Euclidean
Distance weight: Equal
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA disabled

Misclassification Costs
Cost matrix: default

● Number of observations
○ True Positive Rates (TPR) False Negative Rates (FNR)
○ Positive Predictive Values (PPV) False Discovery Rates (FDR)

Model 1

True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	53	7	1	3	1	2	1	3		1					3
airport	1	55		2		2	1	5		1	5			2	1
baseball_diamond	1	4	46	5		1		3	1	1	7	2		1	3
basketball_court	2	4	11	38		3		8			6	1		1	1
beach					61	5				1			6	2	
bridge	1	5		4		55				1	1			4	4
chaparral							74							1	
church	1	2	1	4				44			11	10			2
circular_farmland									73				1	1	
cloud										72			1	2	
commercial_area		4		1				6			64				
dense_residential		1	5	1				3			5	59			1
desert		1					2			4			67	1	
forest					1		1							73	
freeway	2	2	2	3	1	3		2			2	1		4	53

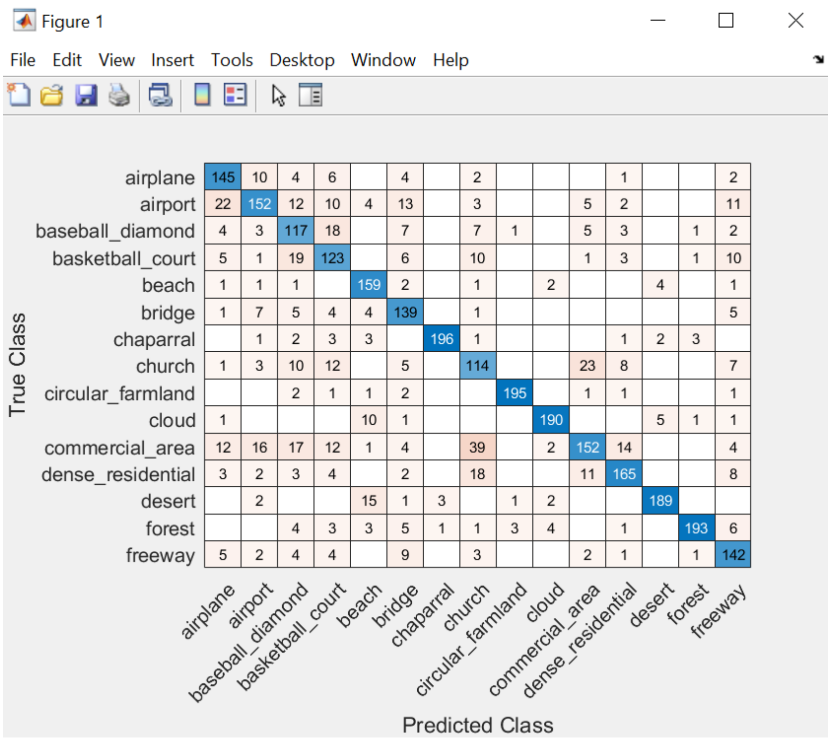
Student Name: Vlad Dubrovenski, Student ID: 16281273

KNN Base Line Testing:

Accuracy:

```
0.8763
```

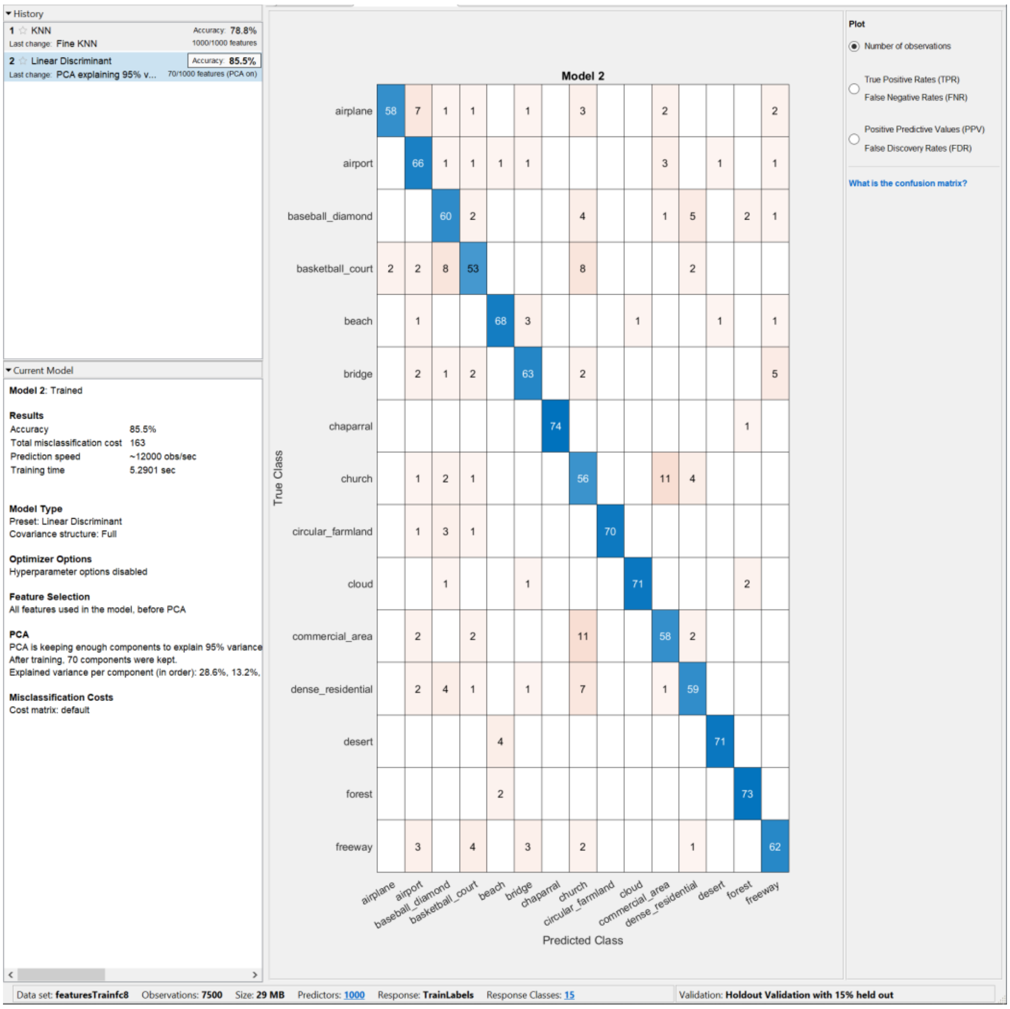
Affinity Map:



	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	145	10	4	6		4		2				1			2
airport	22	152	12	10	4	13		3			5	2			11
baseball_diamond	4	3	117	18		7		7	1		5	3		1	2
basketball_court	5	1	19	123		6		10			1	3		1	10
beach	1	1	1		159	2		1		2			4		1
bridge	1	7	5	4	4	139		1							5
chaparral		1	2	3	3		196	1				1	2	3	
church	1	3	10	12		5		114			23	8			7
circular_farmland			2	1	1	2			195		1	1			1
cloud	1				10	1				190			5	1	1
commercial_area	12	16	17	12	1	4		39		2	152	14			4
dense_residential	3	2	3	4		2		18			11	165			8
desert		2			15	1	3		1	2			189		
forest			4	3	3	5	1	1	3	4		1		193	6
freeway	5	2	4	4		9		3			2	1		1	142

LDA+PCA Validation Accuracy + Confusion Matrix + other information

Student Name: Vlad Dubrovenski, Student ID: 16281273



▾ History

1 ☆ KNN Accuracy: 78.8%
Last change: Fine KNN 1000/1000 features

2 ☆ Linear Discriminant Accuracy: 85.5%
Last change: PCA explaining 95% v... 70/1000 features (PCA on)

▾ Current Model

Model 2: Trained

Results

Accuracy	85.5%
Total misclassification cost	163
Prediction speed	~12000 obs/sec
Training time	5.2901 sec

Model Type
Preset: Linear Discriminant
Covariance structure: Full

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA is keeping enough components to explain 95% variance
After training, 70 components were kept.
Explained variance per component (in order): 28.6%, 13.2%,

Misclassification Costs
Cost matrix: default

Model 2

True Class / Predicted Class

	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	58	7	1	1		1		3			2				2
airport		66	1	1	1	1					3		1		1
baseball_diamond			60	2				4			1	5		2	1
basketball_court	2	2	8	53				8				2			
beach		1			68	3				1			1		1
bridge		2	1	2		63		2							5
chaparral							74							1	
church		1	2	1				56			11	4			
circular_farmland		1	3	1					70						
cloud			1			1				71				2	
commercial_area		2		2				11			58	2			
dense_residential		2	4	1		1		7			1	59			
desert					4								71		
forest					2									73	
freeway		3		4		3		2				1			62

Plot

● Number of observations

○ True Positive Rates (TPR)
False Negative Rates (FNR)

○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)

What is the confusion matrix?

Data set: **featuresTrainfc8** Observations: **7500** Size: **29 MB** Predictors: 1000 Response: **TrainLabels** Response Classes: 15 Validation: **Holdout Validation with 15% held out**

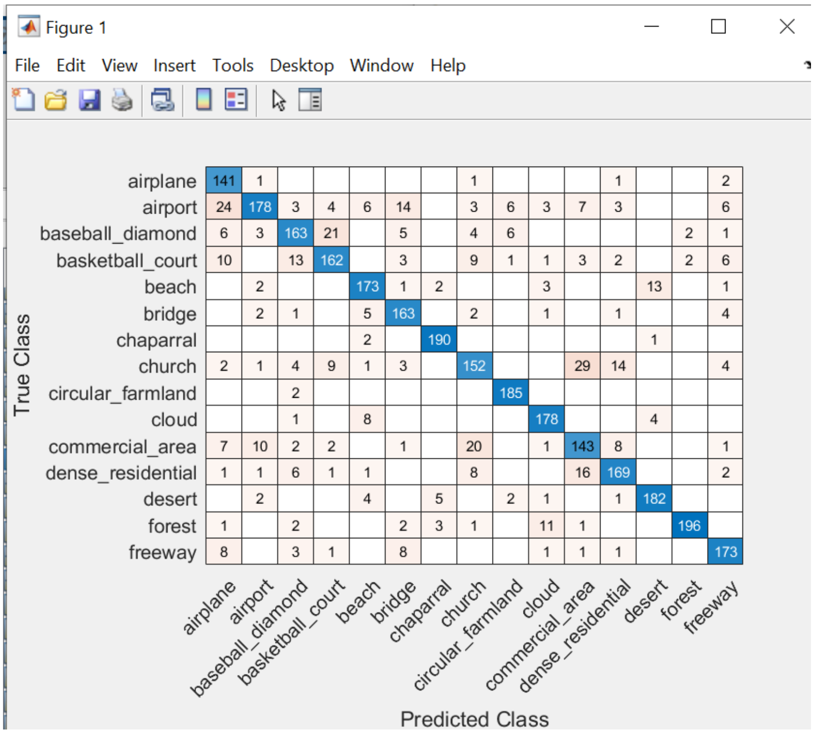
LDA+PCA Testing

Accuracy:

```
0.8493
```

Affinity Map:

Student Name: Vlad Dubrovenski, Student ID: 16281273



	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	141	1						1				1			2
airport	24	178	3	4	6	14		3	6	3	7	3			6
baseball_diamond	6	3	163	21		5		4	6					2	1
basketball_court	10		13	162		3		9	1	1	3	2		2	6
beach		2			173	1	2			3			13		1
bridge		2	1		5	163		2		1		1			4
chaparral					2		190						1		
church	2	1	4	9	1	3		152			29	14			4
circular_farmland			2						185						
cloud			1		8					178			4		
commercial_area	7	10	2	2		1		20		1	143	8			1
dense_residential	1	1	6	1	1			8			16	169			2
desert		2			4		5		2	1		1	182		
forest	1		2			2	3	1		11	1			196	
freeway	8		3	1		8				1	1	1			173

As we can see, the model with fc7 LDA+PCA performed the best.

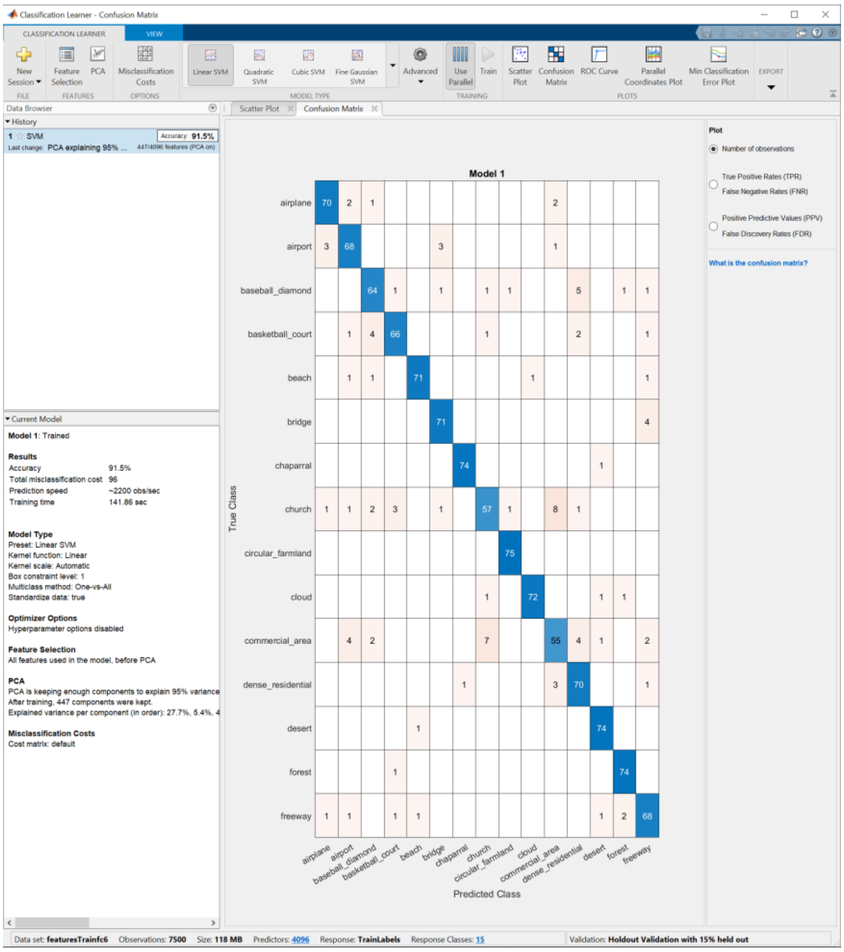
(2) Use (PCA +) Laplacian embedding and 1 vs the rest SVM and compute the top-1 accuracy for fc1, fc2 and fc3, find out which combination gives the best results. (50pts)

To accomplish this task, I again used the Classification Learner to speed up the development. I translated all the code as well. The was one vs all SVM, and I choose linear SVM with added PCA. With a slight modification, I was able to add the Laplacian as well. Below are the results.

FC1

Validation:

Student Name: Vlad Dubrovenski, Student ID: 16281273



History

1 SVM Accuracy: 91.5%
Last change: PCA explaining 95% ... 447/4096 features (PCA on)

Current Model

Model 1: Trained

Results

Accuracy	91.5%
Total misclassification cost	96
Prediction speed	~2200 obs/sec
Training time	141.86 sec

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-All
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA is keeping enough components to explain 95% variance
After training, 447 components were kept.
Explained variance per component (in order): 27.7%, 5.4%, 4

Misclassification Costs
Cost matrix: default

Model 1

True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	70	2	1								2				
airport	3	68				3					1				
baseball_diamond			64	1		1		1	1			5		1	1
basketball_court		1	4	66				1				2			1
beach		1	1		71					1					1
bridge						71									4
chaparral							74						1		
church	1	1	2	3		1		57	1		8	1			
circular_farmland									75						
cloud								1		72			1	1	
commercial_area		4	2					7			55	4	1		2
dense_residential							1				3	70			1
desert					1								74		
forest				1										74	
freeway	1	1		1	1								1	2	68

Plot
● Number of observations
○ True Positive Rates (TPR)
False Negative Rates (FNR)
○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)
What is the confusion matrix?

Data set: featuresTrainfc6 Observations: 7500 Size: 118 MB Predictors: 4096 Response: TrainLabels Response Classes: 15 Validation: Holdout Validation with 15% held out

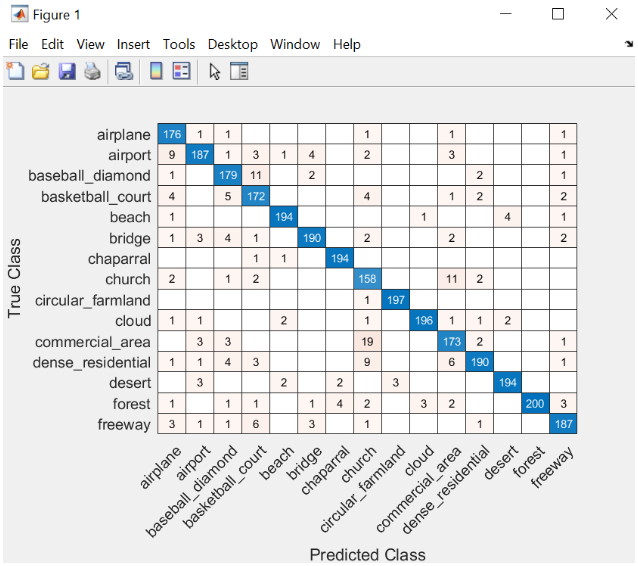
Testing:

Accuracy:

```
0.9290
```

Affinity Map:

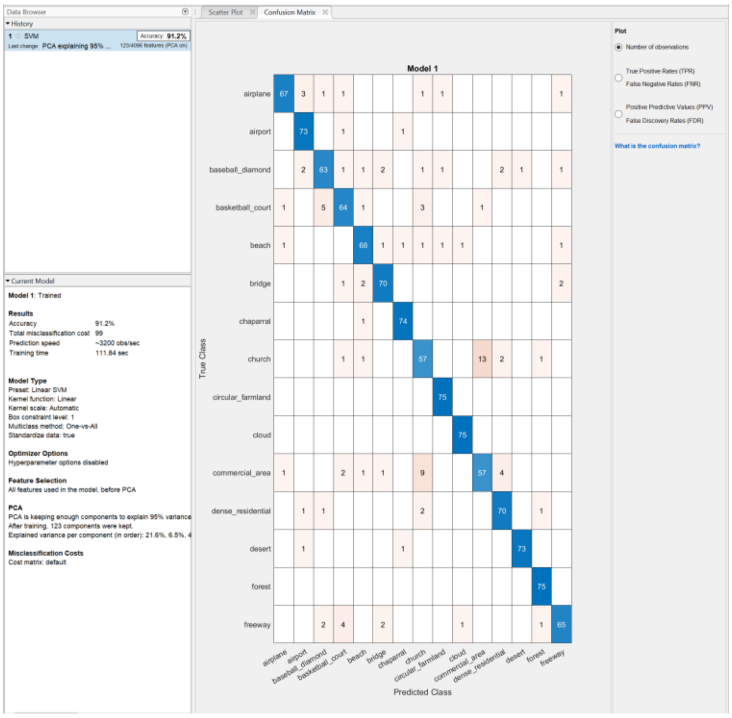
Student Name: Vlad Dubrovenski, Student ID: 16281273



	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	176	1	1					1			1				1
airport	9	187	1	3	1	4		2			3				1
baseball_diamond	1		179	11		2						2			1
basketball_court	4		5	172				4			1	2			2
beach	1				194					1			4		1
bridge	1	3	4	1		190		2			2				2
chaparral				1	1		194								
church	2		1	2				158			11	2			
circular_farmland								1	197						
cloud	1	1			2			1		196	1	1	2		
commercial_area		3	3					19			173	2			1
dense_residential	1	1	4	3				9			6	190			1
desert		3			2		2		3				194		
forest	1		1	1		1	4	2		3	2			200	3
freeway	3	1	1	6		3		1				1			187

FC2

Validation:



1 ☆ SVM Accuracy: 91.2%

Last change: PCA explaining 95% ... 123/4096 features (PCA on)

Model 1: Trained

Results

Accuracy 91.2%

Total misclassification cost 99

Prediction speed ~3200 obs/sec

Training time 111.84 sec

Model Type

Preset: Linear SVM

Kernel function: Linear

Kernel scale: Automatic

Box constraint level: 1

Multiclass method: One-vs-All

Standardize data: true

Optimizer Options

Hyperparameter options disabled

Feature Selection

All features used in the model, before PCA

PCA

PCA is keeping enough components to explain 95% variance

After training, 123 components were kept.

Explained variance per component (in order): 21.6%, 6.5%, 4

Misclassification Costs

Cost matrix: default

Plot

● Number of observations

○ True Positive Rates (TPR)
False Negative Rates (FNR)

○ Positive Predictive Values (PPV)
False Discovery Rates (FDR)

What is the confusion matrix?

	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	67	3	1	1				1	1						1
airport		73		1			1								
baseball_diamond		2	63	1	1	2		1	1			2	1		1
basketball_court	1		5	64	1			3			1				
beach	1				68	1	1	1	1	1					1
bridge				1	2	70									2
chaparral					1		74								
church				1	1			57			13	2		1	
circular_farmland									75						
cloud										75					
commercial_area	1			2	1	1		9			57	4			
dense_residential		1	1					2				70		1	
desert		1					1						73		
forest														75	
freeway			2	4		2				1				1	65

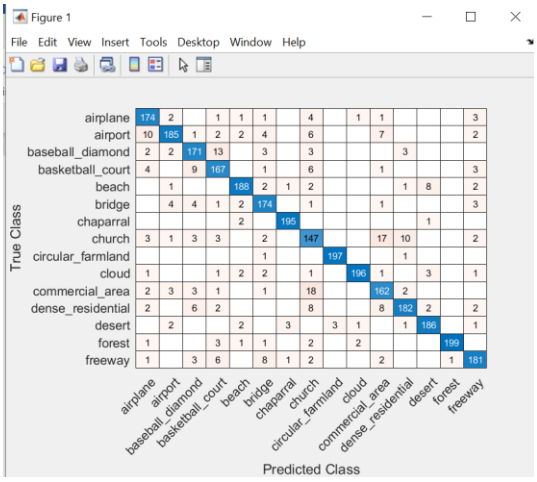
Testing:

Accuracy:

Student Name: Vlad Dubrovenski,   Student ID: 16281273

```
0.9013
```

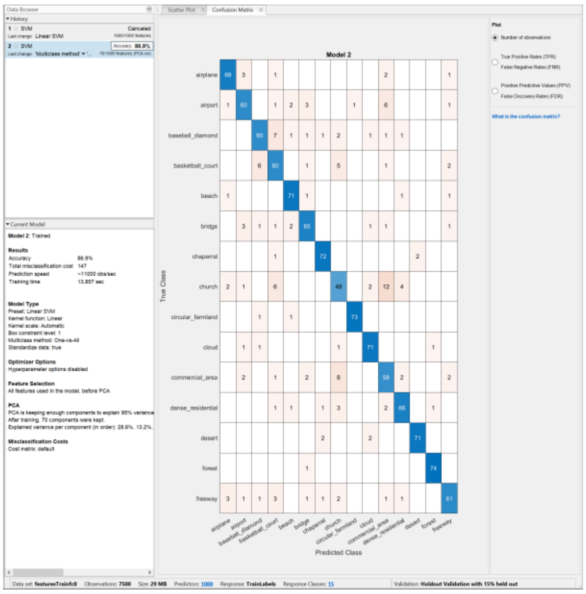
Affinity Map:



	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	174	2		1	1	1		4		1	1				3
airport	10	185	1	2	2	4		6			7				2
baseball_diamond	2	2	171	13		3		3				3			
basketball_court	4		9	167		1		6			1				3
beach		1			188	2	1	2				1	8		2
bridge		4	4	1	2	174		1			1				3
chaparral					2		195						1		
church	3	1	3	3		2		147			17	10			2
circular_farmland						1			197			1			
cloud	1			1	2	2		1		196	1		3		1
commercial_area	2	3	3	1		1		18			162	2			
dense_residential	2		6	2				8			8	182	2		2
desert		2			2		3		3	1		1	186		1
forest	1			3	1	1		2		2				199	
freeway	1		3	6		8	1	2			2			1	181

FC3

Validation:


Data Browser
History
Scatter Plot
Confusion Matrix
Plot
● Number of observations
○ True Positive Rates (TPR) False Negative Rates (FNR)
○ Positive Predictive Values (PPV) False Discovery Rates (FDR)
What is the confusion matrix?
Model 2
True Class
Predicted Class

1 ☆ SVM Canceled
Last change: Linear SVM 1000/1000 features

2 ☆ SVM Accuracy: 86.9%
Last change: 'Multiclass method' = '... 70/1000 features (PCA on)

Current Model

Model 2: Trained

Results

Accuracy	86.9%
Total misclassification cost	147
Prediction speed	~11000 obs/sec
Training time	13.857 sec

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-All
Standardize data: true

Optimizer Options
Hyperparameter options disabled

Feature Selection
All features used in the model, before PCA

PCA
PCA is keeping enough components to explain 95% variance
After training, 70 components were kept.
Explained variance per component (in order): 28.6%, 13.2%,

Misclassification Costs
Cost matrix: default

	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	68	3		1							2				1
airport	1	60		1	2	3			1		6				1
baseball_diamond			60	7	1	1	1	2		1	1	1			
basketball_court			6	60		1		5			1				2
beach	1				71	1						1			1
bridge		3	1	1	2	65				1	1				1
chaparral				1			72						2		
church	2	1		6				48		2	12	4			
circular_farmland			1		1				73						
cloud		1	1					1		71				1	
commercial_area		2		1		2		8			58	2			2
dense_residential				1	1		1	3			2	66		1	
desert							2			2			71		
forest						1								74	
freeway	3	1	1	3		1	1	2			1	1			61

Data set: **featuresTrainfc8** Observations: **7500** Size: **29 MB** Predictors: 1000 Response: **TrainLabels** Response Classes: 15 Validation: **Holdout Validation with 15% held out**

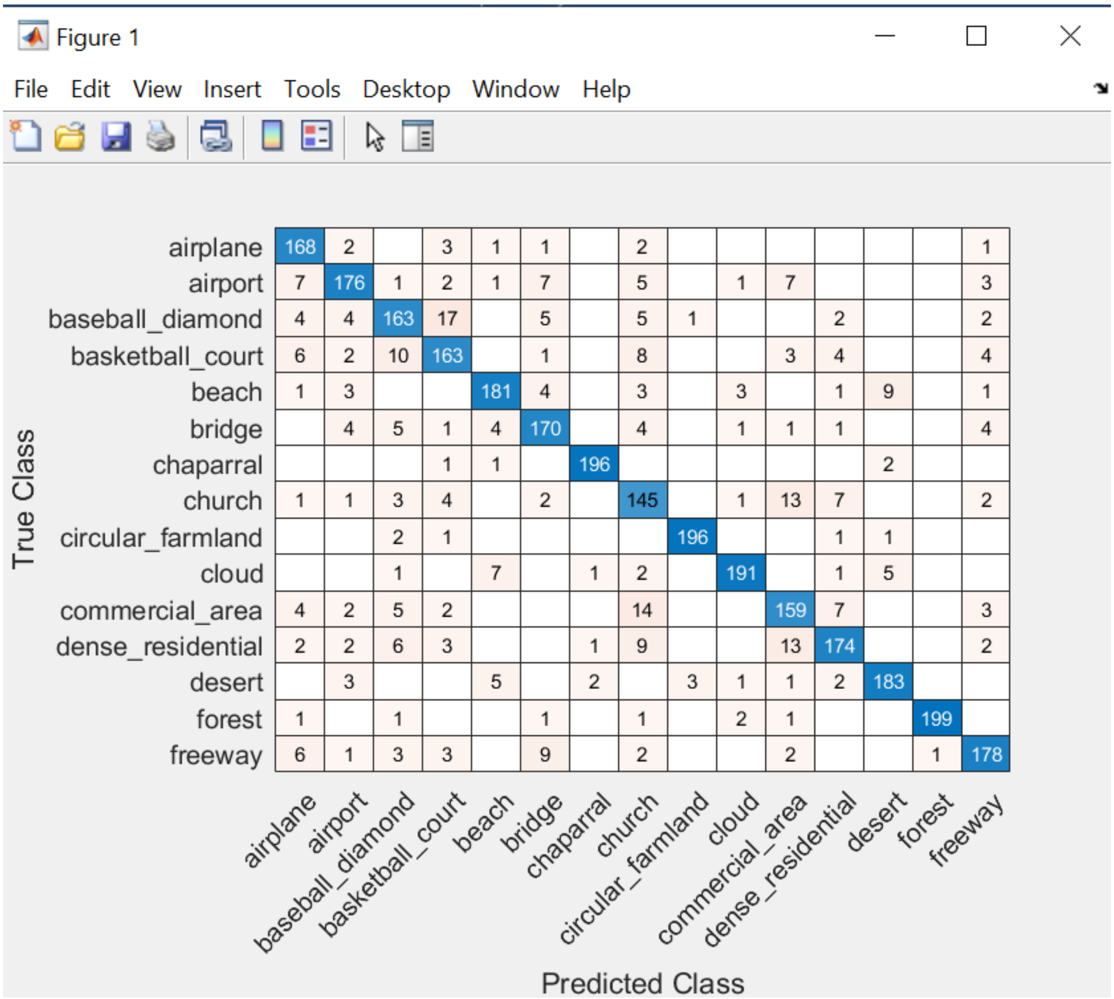
Testing:

Accuracy:

Student Name: Vlad Dubrovenski, Student ID: 16281273

```
0.8807
```

Affinity Map:



True Class \ Predicted Class	airplane	airport	baseball_diamond	basketball_court	beach	bridge	chaparral	church	circular_farmland	cloud	commercial_area	dense_residential	desert	forest	freeway
airplane	168	2		3	1	1		2							1
airport	7	176	1	2	1	7		5		1	7				3
baseball_diamond	4	4	163	17		5		5	1			2			2
basketball_court	6	2	10	163		1		8			3	4			4
beach	1	3			181	4		3		3		1	9		1
bridge		4	5	1	4	170		4		1	1	1			4
chaparral				1	1		196						2		
church	1	1	3	4		2		145		1	13	7			2
circular_farmland			2	1					196			1	1		
cloud			1		7		1	2		191		1	5		
commercial_area	4	2	5	2				14			159	7			3
dense_residential	2	2	6	3			1	9			13	174			2
desert		3			5		2		3	1	1	2	183		
forest	1		1			1		1		2	1			199	
freeway	6	1	3	3		9		2			2			1	178

As we can see, SVM+PCA with FC1 showed the best result out of all competitors.

Running instructions:

Requirements: Matlab 2020b, Deep Learning license, Statistics license, and discrete video card is preferable.

Student Name: Vlad Dubrovenski, Student ID: 16281273

1. Run the splitter.
2. Import vgg16 pretrained. Most likely, you don’t have that model downloaded, so just use the link from the error message. After that, just click on it and follow downloading instruction. You might need an account, it doesn’t have to be attached to a license (gmail worked for me). You may see the architecture to make sure it is the correct model.
3. Extract the features (need some toolbox license, not sure which one, most likely DL again).
4. Run models to see the results of testing. You may also print the validation scores if you would like.