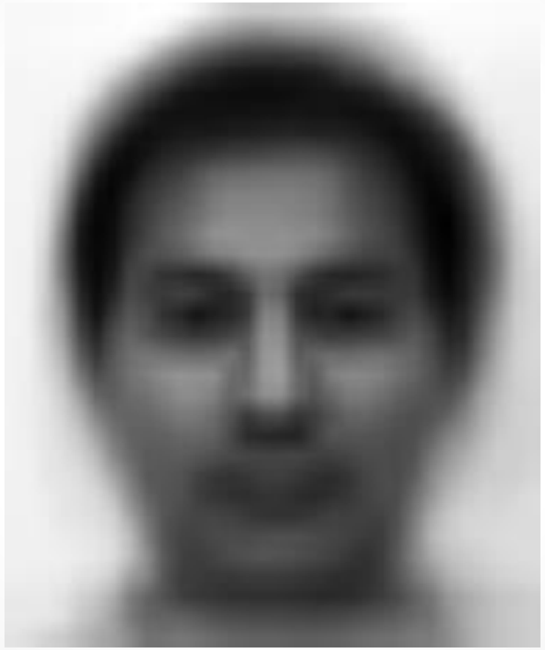
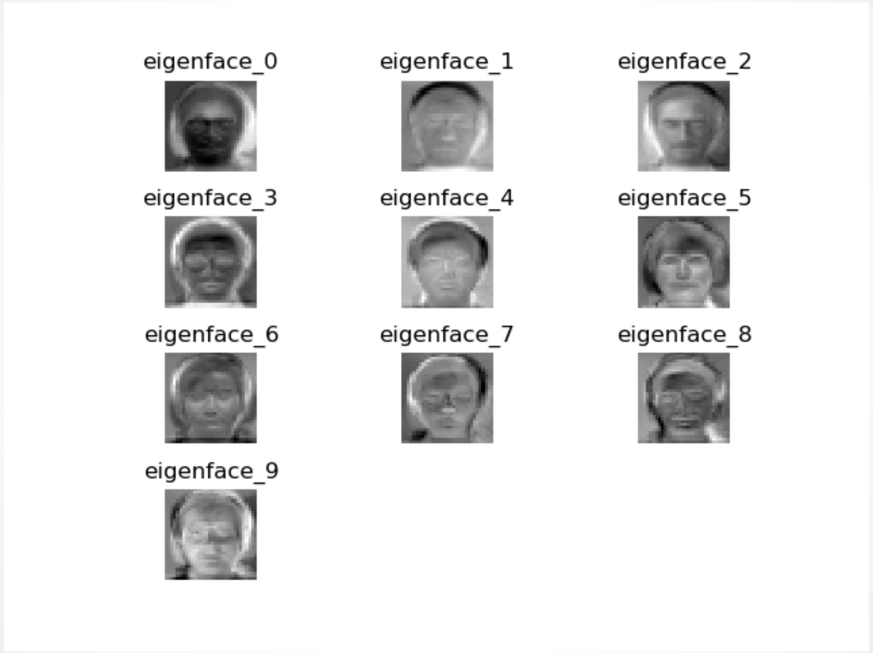
1a: Average face



ps6-1-a-1.png

1b: Eigenvectors



ps6-1-b-1.png

1c: Analysis

	P = 0.2	P = 0.5	P = 0.8	P = 0.95
K = 2	27.3%	31.7%	42.4%	12.5%
K = 5	43.2%	57.3%	66.7%	50%
K = 10	67.4%	75.6%	87.9%	75%

-- From the table above, we can see that these predictions outperform than randomly selecting label between 1 and 15.

-- With K increases, the accuracy also increases. This is expected since more training data are remaining to make the model (eigen vectors) more robust and unbiased.

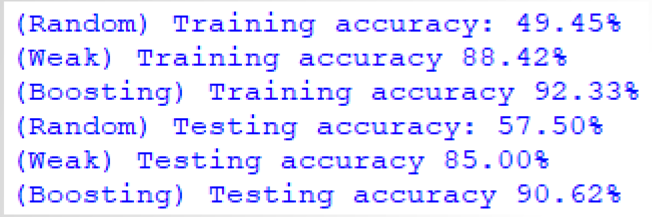
1c: Analysis

	P = 0.2	P = 0.5	P = 0.8	P = 0.95
K = 2	27.3%	31.7%	42.4%	12.5%
K = 5	43.2%	57.3%	66.7%	50%
K = 10	67.4%	75.6%	87.9%	75%

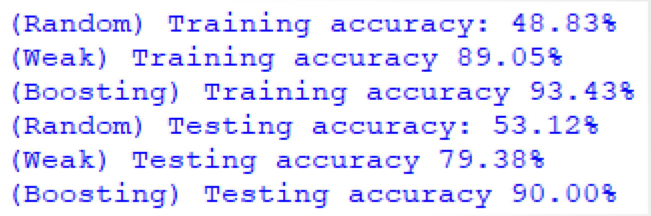
-- With given K, the accuracy improves as the split percentage increases.

-- However, when p is very large (0.95), the accuracy decreases. Thus, proper split percentage is supposed to be within the range [0.5, 0.8], which means training data should be more than testing data and testing dataset should not be too small.

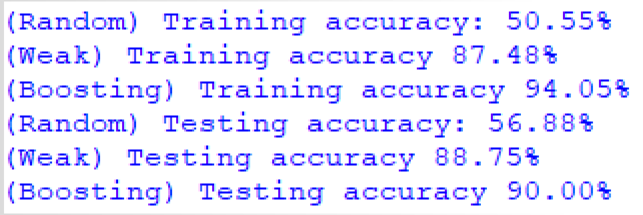
2a: Average accuracy



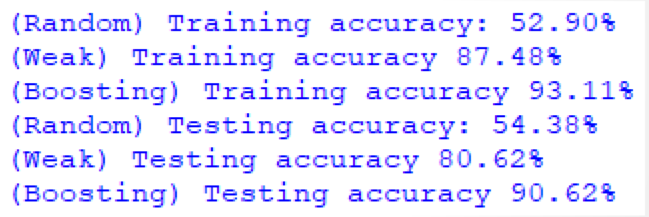
```
(Random) Training accuracy: 49.45%
(Weak) Training accuracy 88.42%
(Boosting) Training accuracy 92.33%
(Random) Testing accuracy: 57.50%
(Weak) Testing accuracy 85.00%
(Boosting) Testing accuracy 90.62%
```



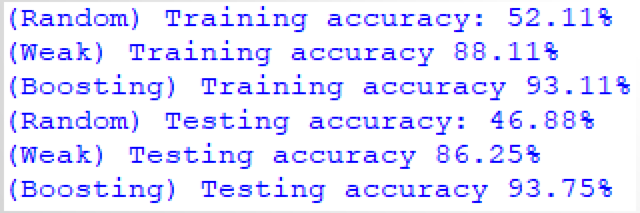
```
(Random) Training accuracy: 48.83%
(Weak) Training accuracy 89.05%
(Boosting) Training accuracy 93.43%
(Random) Testing accuracy: 53.12%
(Weak) Testing accuracy 79.38%
(Boosting) Testing accuracy 90.00%
```



```
(Random) Training accuracy: 50.55%
(Weak) Training accuracy 87.48%
(Boosting) Training accuracy 94.05%
(Random) Testing accuracy: 56.88%
(Weak) Testing accuracy 88.75%
(Boosting) Testing accuracy 90.00%
```



```
(Random) Training accuracy: 52.90%
(Weak) Training accuracy 87.48%
(Boosting) Training accuracy 93.11%
(Random) Testing accuracy: 54.38%
(Weak) Testing accuracy 80.62%
(Boosting) Testing accuracy 90.62%
```



```
(Random) Training accuracy: 52.11%
(Weak) Training accuracy 88.11%
(Boosting) Training accuracy 93.11%
(Random) Testing accuracy: 46.88%
(Weak) Testing accuracy 86.25%
(Boosting) Testing accuracy 93.75%
```

Average (Random) Training accuracy: **50.77%**

Average (Weak) Training accuracy: **88.11%**

Average (Boosting) Training accuracy: **93.21%**

Average (Random) Testing accuracy: **53.75%**

Average (Weak) Testing accuracy: **84%**

Average (Boosting) Testing accuracy: **91%**

2a: Analysis

Random Training Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	50%	49.5%	54.6%
Iter = 10	50.6%	50.5%	50.8%
Iter = 50	50.6%	50.8%	51.6%

Weak Training Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	89.4%	90%	86.5%
Iter = 10	84.4%	89%	88.7%
Iter = 50	86.9%	87.5%	87.3%

Boosting Training Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	89.3%	90%	86.5%
Iter = 10	98.1%	97.8%	97.5%
Iter = 50	100%	100%	100%

Random Testing Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	53.5%	48.6%	49.4%
Iter = 10	48.7%	49.1%	48.8%
Iter = 50	51.9%	47.1%	41.9%

Weak Testing Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	86.2%	85.5%	89.4%
Iter = 10	88.6%	85%	83.8%
Iter = 50	85.6%	86.5%	85.6%

Boosting Testing Accuracy

	P = 0.2	P = 0.5	P = 0.8
Iter = 1	86.2%	85.5%	89.4%
Iter = 10	95.2%	94.2%	93.1%
Iter = 50	98.8%	99.2%	99.4%

2a: Analysis

-- Accuracy: Boosting > Weak Classifier > Random
There is indeed an improvement when using boosting. This is expected since boosting is the linear combination of weak classifier. Weak classifier just pick the best threshold (with minimized errors) to label the data. Thus, when the data is not linear separable, there are always errors when using the weak classifier.

-- When number of iterations increase:
(1) the accuracy of boosting increases while the accuracies for the rest two classifiers did not improve. This is also expected since weak classifier and random classifier is totally independent with the number of iterations.
(2) However, boosting update weights through multiple iterations so that accuracy increases correspondingly.

2a: Analysis

-- When split percentage increases:

(1) the training accuracy and testing accuracy for random classifier did not change.

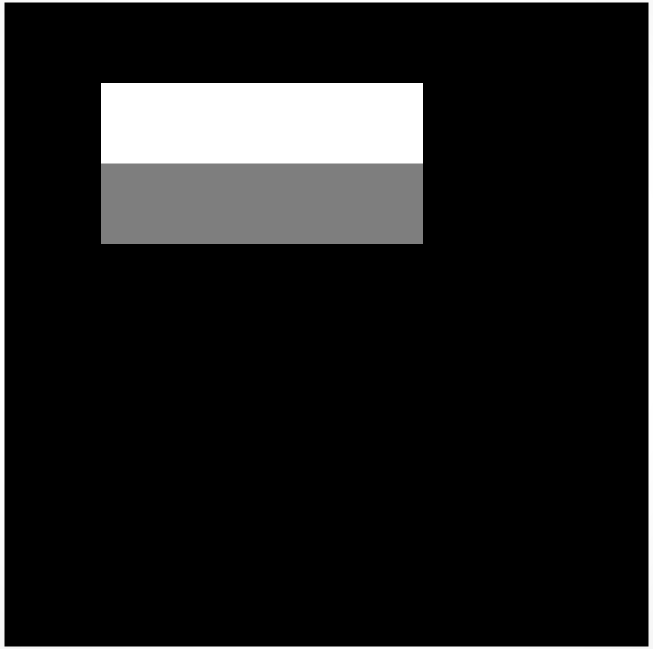
This is expected since the amount of training and testing data has nothing to do with the accuracy if we use the random classifier since there is no 'actual' model learning from the data, what all we do is just random guess.

(2) the training accuracy for weak classifier decreases and testing accuracy for weak classifier increases.

(3) the training accuracy for boosting decreases and testing accuracy for boosting increases.

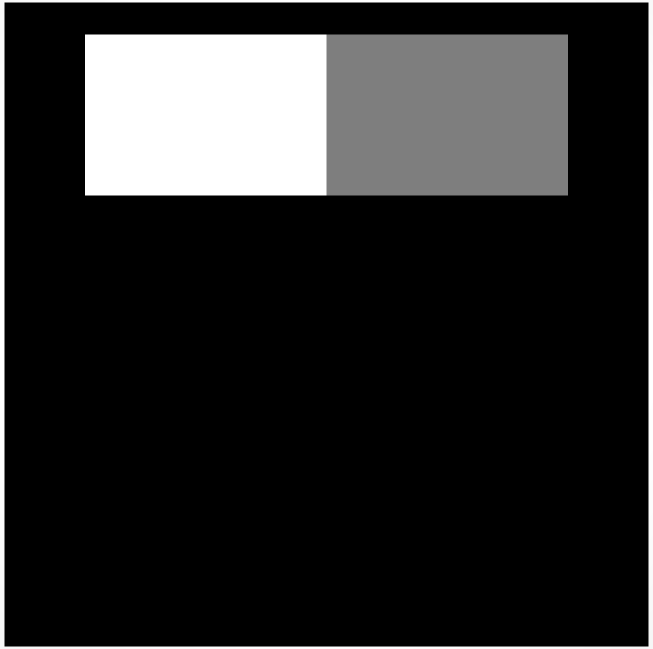
This is expected since more training data causes the model harder to fit, resulting in more training errors. However, testing accuracy increases since model is more unbiased and robust.

3a: Haar Features



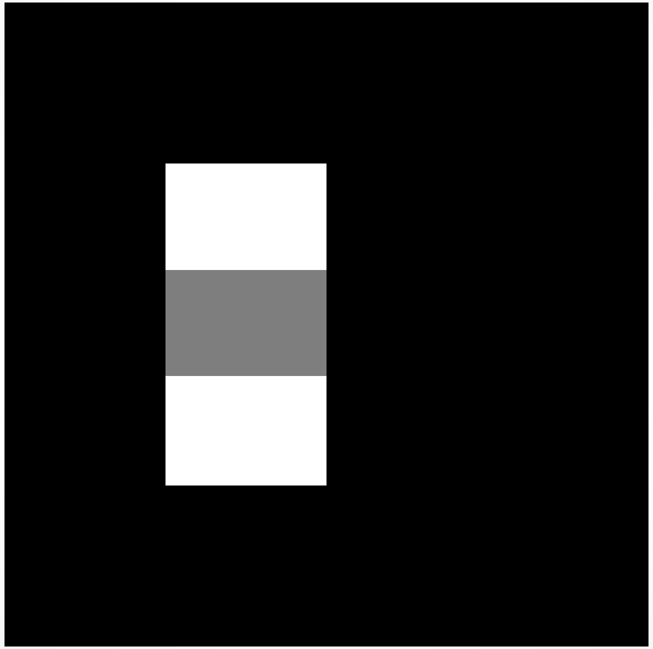
ps6-3-a-1.png

3a: Haar Features



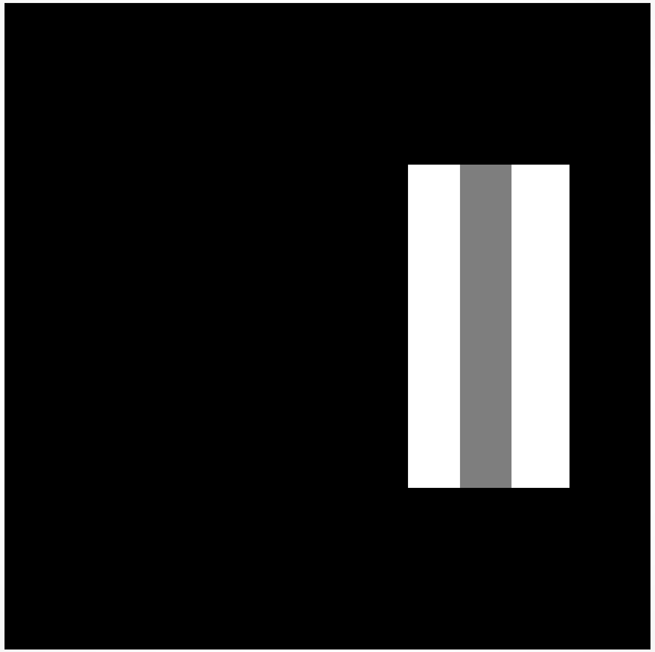
ps6-3-a-2.png

3a: Haar Features

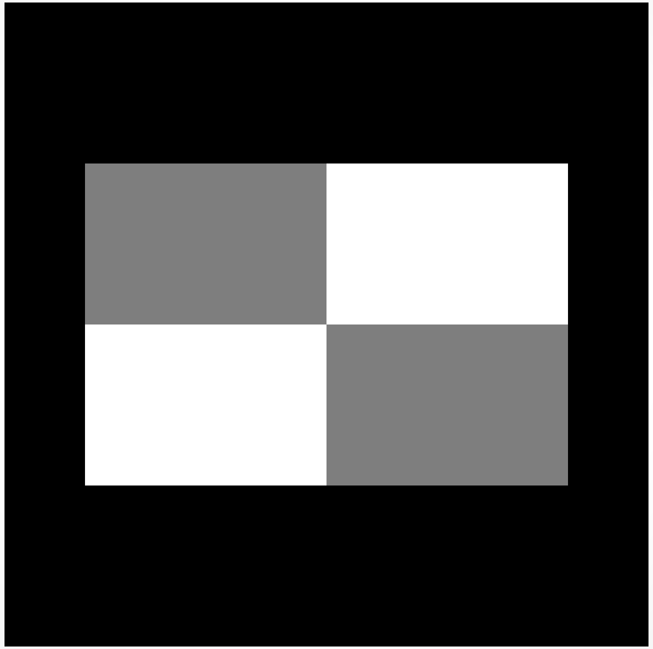


ps6-3-a-3.png

3a: Haar Features


ps6-3-a-4.png

3a: Haar Features



ps6-3-a-5.png

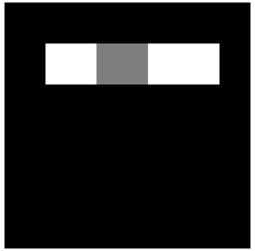
3c: Analysis

How does working with integral images help with computation time? Give some examples comparing this method and np.sum.

Answer: Integral images offers a one-time computation time to compute all integral images of all different sizes. By using the integral images, the computation time is constant for each rectangle sum calculation no matter how large of the rectangle.

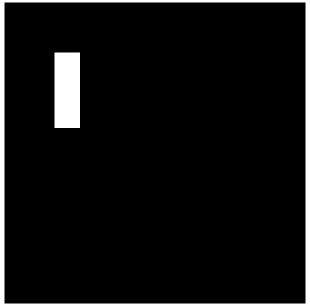
However, if we using np.sum, the computation time is proportional to the number of pixels covered by the rectangle. Also, the total time for the same area rectangle computation by using np.sum is much larger than the integral image approach.

4b: Viola Jones Features



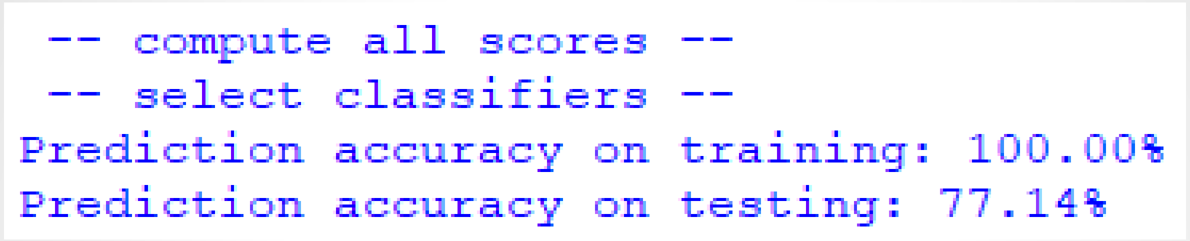
ps6-4-b-1.png

4b: Viola Jones Features



ps6-4-b-2.png

4b: Analysis



```
 -- compute all scores --
 -- select classifiers --
Prediction accuracy on training: 100.00%
Prediction accuracy on testing: 77.14%
```

What do the selected Haar features mean? How do they contribute in identifying faces in an image?

Answer: The first selected Haar feature means the eyes area. (white + black + white) and the second means the nose area. They are the most relevant features of face detection.