# Machine Learning for Computer Vision (EE5177)
# Programming Assignment 3 : Regression Models
# Problem #2

Akshit Kumar
*EE14B127*

29th March 2017

## Contents

# 1 Introduction

## 1.1 Goal

The goal of the problem is to estimate the head pose of a face using various regression techniques like Linear Regression, Dual Regression, Linear Regression with PCA, Dual Regression with Kernel Trick.

## 1.2 Approach

To estimate the head pose of a face, we employ various regression techniques mentioned above on Histogram of Oriented Gradients (HoG) features.

## 1.3 Analysis

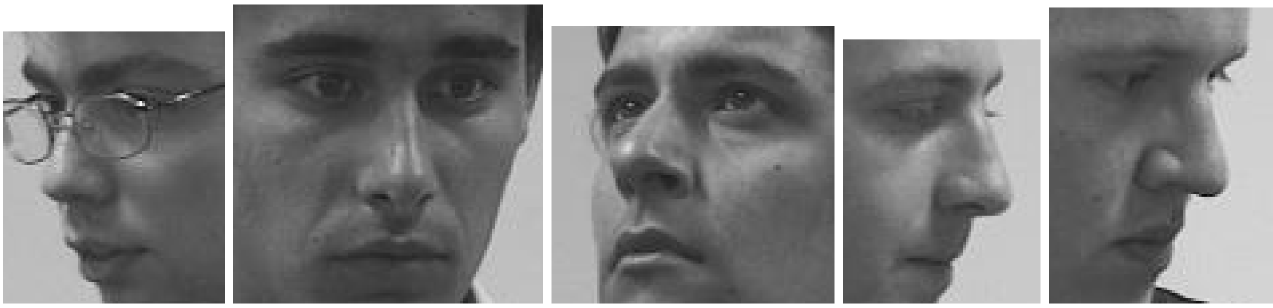
Each regression technique is described here and in conclusion the results obtained from the various regression techniques are evaluted over some test images.

# 2 Feature Extraction

## 2.1 Approach

The actual images are first converted to grayscale images, then they are cropped to only contain the faces and then they are resized to be of the dimension 96px X 108 px. After resizing, the images are used to extract the Histogram of Oriented Gradients feature vectors using the matlab code provided.

## 2.2 Sample Cropped Images



# 3 Linear Regression using HOG features

## 3.1 ML Estimate

The ML estimate of the weights can be calculated using

$$\phi = (XX^T)^{-1}Xw$$

where $X = [x_1, x_2, ........., x_I]$ represents the matrix of HoG features and $w$ represents the world states.

## 3.2 Train and Test Error

Train RMSE = ***1.1778e-11***
Test RMSE = ***13.6378***

## 3.3 Analysis

Clearly the model is *overfitting* the data which is reflected by the fact that the train rmse is too low compared to the test rmse which implies that the model doesn't generalise well. Also this method is computationally intensive as $XX^T$ is not full rank and not calculating inverse, we need to take a pseudo inverse.

## 4 Dual Regression using HOG features

### 4.1 ML Estimate

The ML estimate of the weights can be calculated using

$$\phi = X(X^T X)^{-1} w$$

where $X = [x_1, x_2, ........, x_I]$ represents the matrix of HoG features and $w$ represents the world states. We can show that this estimate is the same as what we would get in the case of linear regression.

### 4.2 Train and Test Error

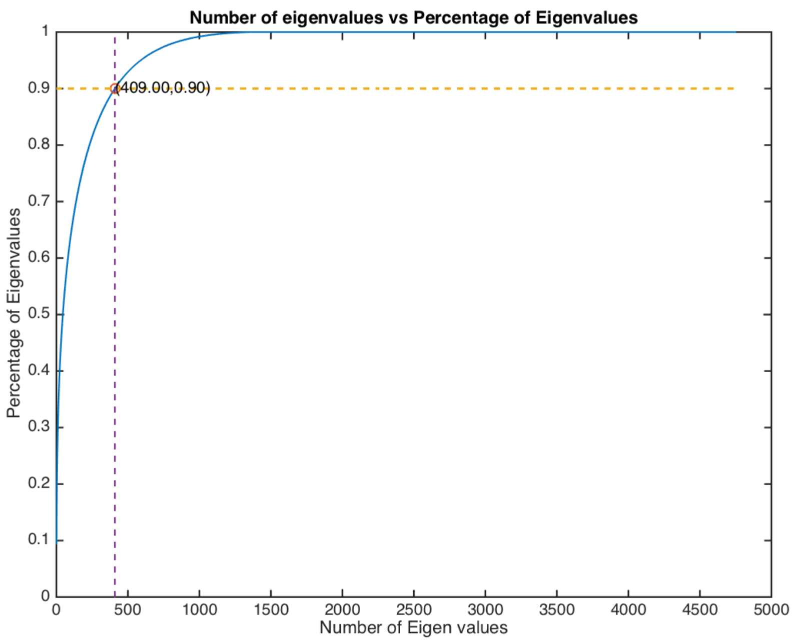
Train RMSE = ***3.5667e-11***
Test RMSE = ***13.6378***

### 4.3 Analysis

Clearly the model is *overfitting* the data which is reflected by the fact that the train rmse is too low compared to the test rmse which implies that the model doesn't generalise well. This method performs as well as the Linear regression methos but is computationally less intensive as $X^T X$ is full rank and can be easily inverted.

## 5 Linear Regression using PCA on HOG features

### 5.1 Choosing the number of components for PCA




### 5.2 Appropriate value of number of components for PCA

At k = 409, the k largest eigenvalues capture 90% of the variance.

### 5.3 Train and Test Error

Train RMSE = ***7.8429***
Test RMSE = ***10.5769***

### 5.4 Analysis

Linear Regression using PCA doesn't suffer from overfitting like linear regression and dual regression as PCA tries to capture patterns in data by projecting the data along the plane of highest variance giving simpler models.

# 6 Dual Regression with Kernel Trick

## 6.1 RBF Kernel

In this case we make use of the RBF Kernel, we replace the dot product of 2 vectors with the kernel function as

$$k[x_i, x_j] = exp[-0.5(\frac{(x_i - x_j)^T(x_i - x_j)}{\lambda^2})]$$

## 6.2 Train and Test Error

### 6.2.1 $\lambda_1$ as the minimum pairwise distance

Train RMSE = ***3.6863e-13***
Test RMSE = ***12.4462***

### 6.2.2 $\lambda_2$ as the maximum pairwise distance

Train RMSE = ***1.4214e-11***
Test RMSE = ***9.6083***

### 6.2.3 $\lambda_3$ as the mean of $\lambda_1$ and $\lambda_2$

Train RMSE = ***2.7439e-12***
Test RMSE = ***8.9713***

## 6.3 Analysis

Again, all the three models suffer from gross *overfitting*. However this model seems to be performing better than all the other models discussed above as the average value of lambdas gives the least training error and hence generalises best compared to other models discussed above.

# 7 Comparison

## 7.1 Comparison of various models in terms of Train and Test RMSE

| Model | Train RMSE | Test RMSE |
|---|---|---|
| Linear Regression | 1.1778e-11 | 13.6378 |
| Dual Regression | 3.5667e-11 | 13.6378 |
| Linear Regression using PCA | 7.8429 | 10.5769 |
| Dual Regression with Kernel Trick ($\lambda_1$) | 3.6863e-13 | 12.4462 |
| Dual Regression with Kernel Trick ($\lambda_2$) | 1.4214e-11 | 9.6083 |
| Dual Regression with Kernel Trick ($\lambda_3$) | 2.7439e-12 | 8.9713 |

## 7.2 Analysis

We can clearly see that Dual Regression with Kernel Trick ($\lambda_3$)seems to perform the best in terms of test rmse metric. We also see that all the models apart from PCA suffer from huge *overfitting* problems and hence it is doesnt generalise well. The reason why Dual Regression with kernel trick does better than other models considered is because these models are able to model the non-linearities in data better which linear models can't really capture. The reason $\lambda_3$ might be doing better than other $\lambda$ values is that it is the mean of the $\lambda_1$ and $\lambda_2$ and is closer to variance than $\lambda_1$ and $\lambda_2$ and hence models the variance better.

## 7.3 Pan Angles For Images

| Model | test/122.jpg | test/337.jpg | test/405.jpg | test/428.jpg | test/550.jpg |
|---|---|---|---|---|---|
| Linear Regression Angle | 74.1186 | -51.1052 | -66.0633 | 29.3632 | -6.8440 |
| Dual Regression Angle | 74.1186 | -51.1052 | -66.0633 | 29.3632 | -6.8440 |
| Linear Regression using PCA Angle | 90.2055 | -53.8175 | -74.1713 | 50.6014 | 3.0786 |
| Dual Regression with Kernel Trick ($\lambda_1$) Angle | 80.4537 | -48.6835 | -72.7297 | 39.0730 | 5.1508 |
| Dual Regression with Kernel Trick ($\lambda_2$) Angle | 82.1382 | -49.0304 | -72.3169 | 44.5357 | -2.9032 |
| Dual Regression with Kernel Trick ($\lambda_3$) Angle | 82.8429 | -48.2168 | -74.1274 | 46.2457 | 0.4380 |
| True Angle | 90 | -45 | -75 | 60 | 0 |

### 7.4 Analysis on Sample Test Images

From the sample test images, we can clearly see that PCA and Dual regression with kernel trick ($\lambda_3$) seems to be best models for estimation of the head pose angle. Also linear and dual regression perform very badly in estimation which is expected as these models don't capture the variance and non-linearities in the data well.