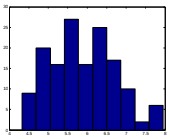
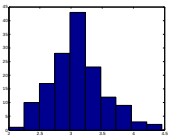
Kyle Benson
CS 273A - Machine Learning: Fall 2013
Homework 1

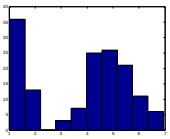
Problem 3:

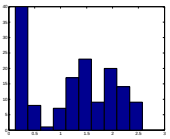
(a) There are 4 features and 148 data points.

(b) Histograms of the 4 features:


First


Second


Third


Fourth

(c) Calling `mean` on each column gives the following values:

feature	one	two	three	four
mean	5.9001	3.0989	3.8196	1.2526

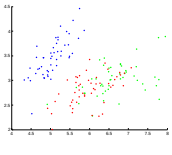
(d) Calling `var` and `stdev` on each feature gives:

feature	one	two	three	four
variance	0.6993	0.1916	3.0976	0.5797
stdev	0.8362	0.4378	1.7600	0.7613

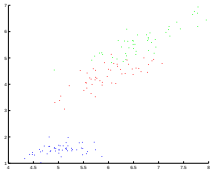
(e) I used the following code to store the normalized data in `d.norm_data`:

```
d.data = iris(:,1:end-1);     % extract data
d.mean = mean(d.data)
d.stdev = std(d.data)
d.norm_data = bsxfun(@minus, d.data, d.mean) % subtract mean
d.norm_data = bsxfun(@rdivide, d.norm_data, d.stdev) % divide by stdev
```

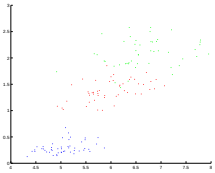
(f) Using different values in the line
`scatter(iris(iris(:,5)==target,1), iris(iris(:,5)==target,feat), [color '.'])`,
I got the following plots:



1v2



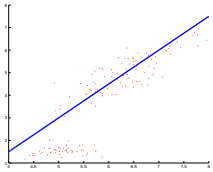
1v3



1v4

Problem 4:

Following the given steps gives this plot:



Problem 5:

Using the following function:

```
function D = dist(x, X)
% Calculates the vector of Euclidean distances between a single vector x
% and a collection of data points (stored in a matrix) X.

differences = bsxfun(@minus, x, X);
squared_differences = differences.^2;
sum_squared_differences = sum(squared_differences, 2);
D = sqrt(sum_squared_differences);

end
```

I executed the command `plot(mydist(iris(1,:), iris))` to get the following plot: