# Problem 1

1. 0.1 \* 30/37 + 0.3\*1/2 + 0.6\*3/10 = 0.4111
2. P(orange | red box) = 0.0108 / (0.0108 + 0.15 + 0.18) = 0.0317

P(red box | orange) = 0.0317 \* 0.1 / 0.3408 = 0.0093

|  |  |  |  |
| --- | --- | --- | --- |
|  | r (0.1) | b (0.3) | g (0.6) |
| apple | 0.0811 | 0.15 | 0.18 |
| orange | 0.0108 | 0.15 | 0.18 |
| lime | 0.0081 | 0 | 0.24 |

# Problem 2

1. Expected value = p/6 + 2p/6 + 3p/6 + 4p/6 +5p/6 + 6 – 5p = 6 – 5p/2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| Heads (p) | p/6 | p/6 | p/6 | p/6 | p/6 | p/6 |
| Tails (1-p) | 0 | 0 | 0 | 0 | 0 | 1-p |

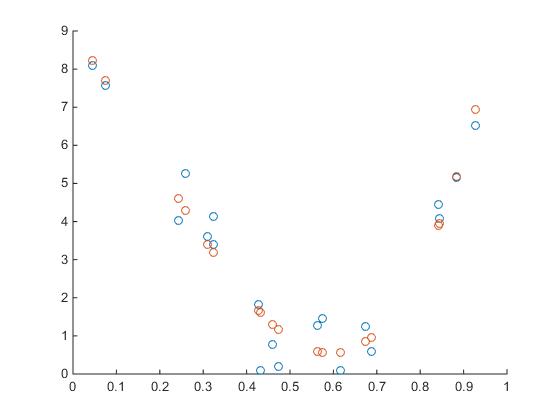
1. Variance = p/6 + 4p/6 + 9p/6 + 16p/6 + 25p/6 + 36 – 30p – (6 – 5p/2)   
   = 36 – 125p/6 – (36 – 30p + 25p2/4)   
   = 36 – 125p/6 – 36 + 30p -25p2/4  
   = 55p/6 – 25p2/4

# Problem 3

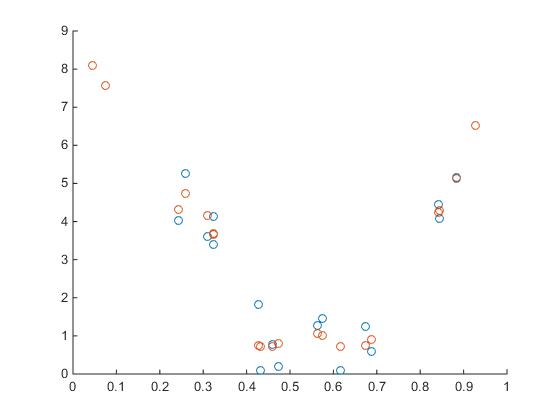


|  |  |  |
| --- | --- | --- |
| **Degree \\ Dataset** | **Training** | **Test** |
| **1** | 2.3668 | 2.9984 |
| **2** | 0.8451 | 0.9768 |
| **3** | 0.6362 | 0.7060 |
| **4** | 0.5921 | 0.6255 |
| **5** | 0.5918 | 0.6251 |
| **6** | 0.5849 | 0.6024 |
| **7** | 0.5678 | 0.5847 |
| **8** | 0.5448 | 0.5803 |
| **9** | 0.4542 | 0.5778 |
| **10** | 0.4418 | 0.5094 |

1. blue is training, orange is fitted



1. blue is training, orange is fitted



1. see next page

% produce errors and plot

function linbasereg(power)

load Xtrain;

load Ytrain;

load Xtest;

load Ytest;

%%% training %%%

p = phi(power, Xtrain);

fw = calcfw(p, Ytrain);

% print training errors

disp(sqrt(sum((fw - Ytrain) .^ 2) / 20));

%%% test data %%%

p2 = phi(power, Xtest);

fw2 = calcfw(p2, Ytest);

% get test data errors

disp(sqrt(sum((fw2 - Ytest) .^ 2) / 20));

% plot

figure(gcf);

scatter(Xtrain, Ytrain);

hold on;

scatter(Xtrain, fw);

hold off;

end

% helper to produce phi from given degree and x vector

function mat = phi(degree, xmat)

numcol = degree + 1;

mat = zeros(20, numcol);

for i = 1:degree + 1

mat(:, i) = xmat.^(i - 1);

end

end

% helper to produce fw from given phi and y vector

function mat = calcfw(p, ymat)

mat = p \* ((p' \* p) \ (p' \* ymat));

end