

# Uebungsblatt 4

## „Mustererkennung“

J. Cavojska, N. Lehmann, R. Toudic

01.06.2015

### Inhaltsverzeichnis

<b>1</b>	<b>Aufbereitung der Daten</b>	<b>2</b>
<b>2</b>	<b>Aufgabe 1 (Lineare Regression)</b>	<b>3</b>
<b>3</b>	<b>Aufgabe 2 (Subset Selection)</b>	<b>5</b>

# 1 Aufbereitung der Daten

```
1 % fish.txt: index; the age of the fish; the water temperature in Celsius; ↵  
    the length of the fish  
2 A = load('fish.txt');  
3  
4 % winequality-red.txt: fixed acidity; volatile acidity; citric acid; ↵  
    residual sugar; chlorides; free sulfur dioxide; total sulfur; dioxide; ↵  
    density; pH; sulphates; alcohol; quality (score between 0 and 10)  
5 B = load('winequality-red.txt');
```

## 2 Aufgabe 1 (Lineare Regression)

*Schaetzen Sie den Wert fuer length anhand der Parameter age und temperature mit linearer Regression.*

*Visualisieren Sie dreidimensional die tatsaechlichen Datenpunkte, die geschaezten Datenpunkte, die Ebene auf die projiziert wurde, sowie die Abstaende der tatsaechlichen Datenpunkte zu dieser Ebene.*

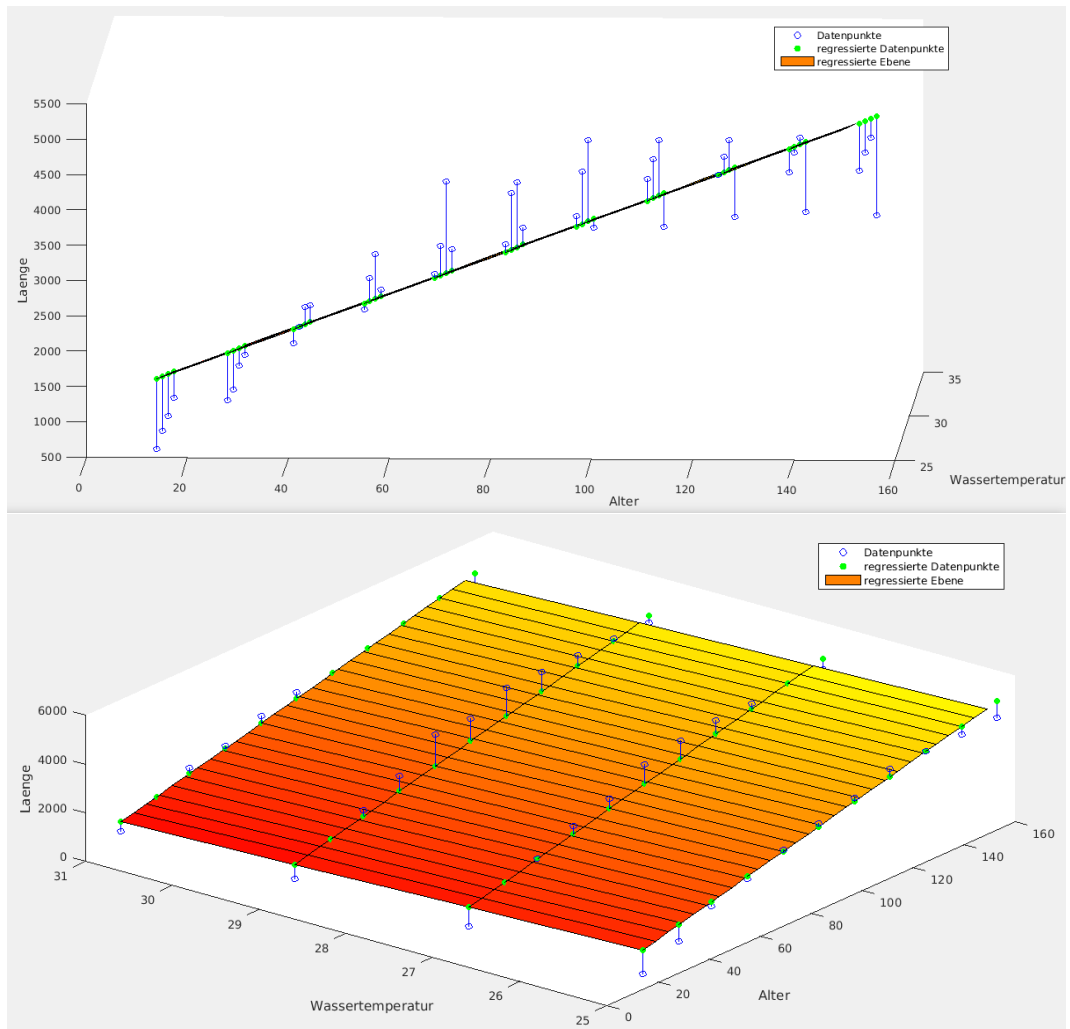
```
1 y = A(:,4); % nur die Laengen der Fische
2 Z = A(:,2:3); % alle Datenpunkte, ausser Laenge
3 onesVector = ones(size(Z,1), 1); % Spaltenvektor mit Einsen der gleichen ←
   Laenge wie A
4 X = horzcat(onesVector, Z); % Einsen-Vektor an Datenpunkte-Matrix ←
   drankleben
5
6 % X' ist die transponierte Matrix X
7 beta = inv(X'*X) * X' * y; % beta ist der Vektor, mit dem Eingabedaten ←
   multipliziert werden muessen, damit wir an die Klassen-Labels kommen
8 % Resultat:
9 % beta =
10 %      1.0e+03 *
11 %
12 %      3.9043
13 %      0.0262
14 %      -0.1064
15
16 % Plot
17 figure('NumberTitle','off','Name','Aufgabe 1 - Mesh');
18 hold on
19
20 % Bildpunkte der Rohdaten
21 Bildpunkte = A(:,2:4);
22 scatter3(Bildpunkte(:,1), Bildpunkte(:,2), Bildpunkte(:,3),30,[0 0 1]);
23
24 % Bildpunkte der geschaezten Daten
25 EstimatedLength = X*beta;
26 % Ebene mit Geschaetzter Laenge
27 EstimatedData = horzcat(Z, EstimatedLength);
28 % Plot
29 scatter3(EstimatedData(:,1), EstimatedData(:,2), EstimatedData(:,3),30,[0 1 ←
   0], 'filled');
30
31 % Intervalle bestimmen
32 min_A = min(A(:,2:4));
33 max_A = max(A(:,2:4));
34
35 x1 = min_A(1,1):5:max_A(1,1);
36 x2 = min_A(1,2):2:max_A(1,2);
37 [D1, D2] = meshgrid(x1,x2);
38 M = beta(1)+beta(2)*D1+beta(3)*D2;
39 surf(D1,D2,M);
40 colormap autumn;
41
```

```

42 % Linien von Datenpunkten auf Ebene einzeichnen
43 for index = 1:length(A(:,4))
44     plot3([Z(index,1) Z(index,1)]', [Z(index,2) Z(index,2)]', [y(index) ←
        EstimatedLength(index)]', '-b');
45 end
46
47 xlabel('Alter');
48 ylabel('Wassertemperatur');
49 zlabel('Laenge');
50 legend('Datenpunkte', 'regressierte Datenpunkte', 'regressierte Ebene')

```

Eine grafische Darstellung der tatsaechlichen und der durch die Regression entstandenen Datenpunkte:



### 3 Aufgabe 2 (Subset Selection)

Schätzen Sie den Wert fuer quality mit linearer Regression anhand aller moeglichen Kombinationen der anderen Parameter (also jeweils fuer alle Einer-, Zweier-, Dreierkombinationen, usw.) und berechnen jeweils die Summe der quadratischen Abweichungen zwischen den geschätzten und tatsaechlichen Werten fuer quality.

Visualisieren Sie dies als zweidimensionalen Plot. Auf der x-Achse steht dabei die Anzahl der verwendeten Parameter, auf der y-Achse die Summe der quadratischen Abweichungen.

```
1 y = B(:,12); % Spalte mit Weinqualitaet
2 Result = []; % Ergebnismatrix
3
4 featureIndices = [1 2 3 4 5 6 7 8 9 10 11];
5 for numFeatures = 1:11 % es gibt 11 features, anhand welcher man ↵
    klassifizieren kann
6     combinations = combnk(featureIndices, numFeatures); % combnk gibt eine ↵
        Liste aller n ueber k vielen Kombinationen von featureIndices ↵
        zurueck (wobei numFeatures unser k ist)
7     for line = 1:size(combinations, 1)
8         combination = combinations(line, :);
9         X = B(:, combination);
10        onesVector = ones(size(X,1), 1); % Spaltenvektor mit Einsen der ↵
            gleichen Laenge wie B
11        X = horzcat(onesVector, X); % Einsen-Vektor an Datenpunkte↵
            Matrix drankleben
12        beta = inv(X'*X) * X' * y; % beta ist der Vektor mit den ↵
            Koeffizienten der Regressionsebene
13
14        Q = (y - X*beta)'*(y - X*beta); % mean squared error
15        Result = vertcat(Result, [numFeatures, Q]);
16    end
17 end % for numFeatures
18
19 figure('NumberTitle','off','Name','Aufgabe 2 - Scatter');
20 scatter(Result(:,1),Result(:,2));
```

