Supervised classification

Karina Nielsen (karni@space.dtu.dk) and Allan Aasbjerg Nielsen

November, 2022

Overview

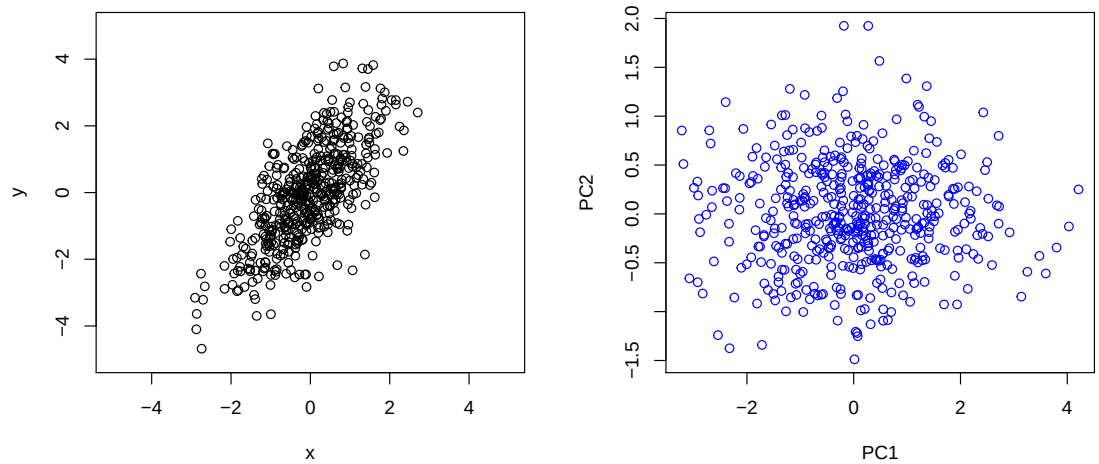
- Repetition from last week PCA
- Classification via Logistic regression
  - Example and exercise
- Probability and Bayes Rule
- Supervised classification: Maximum likelihood classification
  - Examples
- Exercises (Report, not mandatory, but strongly recommended to do)

Repetition of PCA - Intuitive description of PCA

- A linear transform, that transform the correlated variables Z into uncorrelated factors Y

$$Y = ZA$$

- Z is an $n \times m$ matrix where m is the number of features (e.g. bands) and n is the number of observations.
- A is an $m \times m$ matrix



- This means that the covariance matrix of Y must be a diagonal matrix
- Now we just need to find the rotation matrix A

Repetition of PCA - Summary of the PCA transform

- Standardize the d-dimensional dataset
- Construct the data covariance matrix
- Decompose the covariance matrix into its eigenvectors and eigenvalues.
- Sort the eigenvalues by decreasing order to rank the corresponding eigenvectors.
- Construct a projection matrix A from all the (or the "top" k) eigenvectors.
- Transform the m-dimensional input dataset X using the projection matrix A to obtain the new k-dimensional feature subspace, where k is equal to or smaller that m

Repetition of PCA - Interpretation of the eigenvalues and eigenvectors

- The eigenvalues gives the variance in the direction of the eigenvectors
- We can calculate the scores (how much of the total variance they each describe) of the respective eigenvalues

$$score = \frac{\lambda_i \times 100}{\sum_{i=1}^{m}}$$

- The relation between the original bands and the PCs are described via the loads

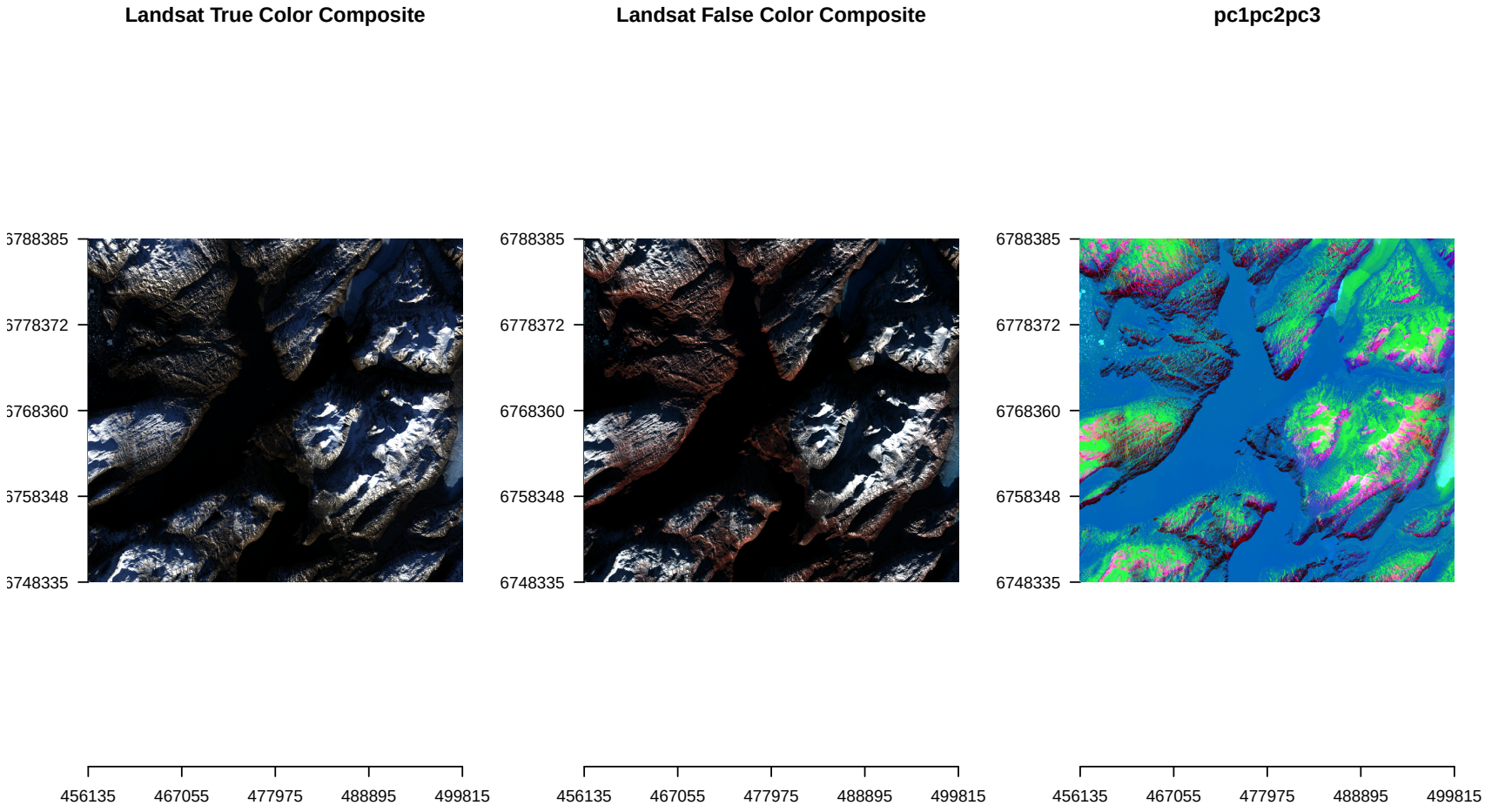
$$R = \frac{a_{km} \times \sqrt{\lambda_m}}{\sqrt{(Var_k)}}$$

- Here a_{km} is eigenvector component related to the band k and PC component m
- λ_m is the mth eigenvalue
- Var_k is the variance of the kth band in the data covariance matrix. If standardized this is 1

Repetition of PCA - main points Comments regarding and applications of PCA

- The direction/sign of the eigenvectors is arbitrary
- The PCA is scale dependent (standardization of the data or not)
- The PCs does not have a direct physical but are linear combinations of the original bands
- Applications
  - Widely used method to reduce the dimension of a data set to lower dimensions for analysis, visualization or data compression.
  - Most variance in the data is typically captured in the first 3 PCs
  - . . . Hence, remove redundancy
  - Enhance detains in image
  - Change detection
  - Classification

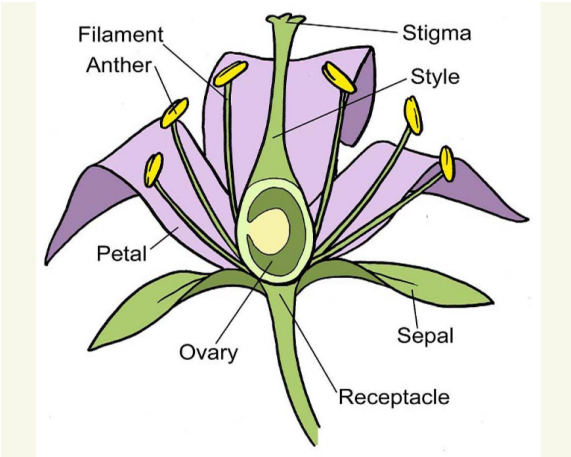
Greenland example



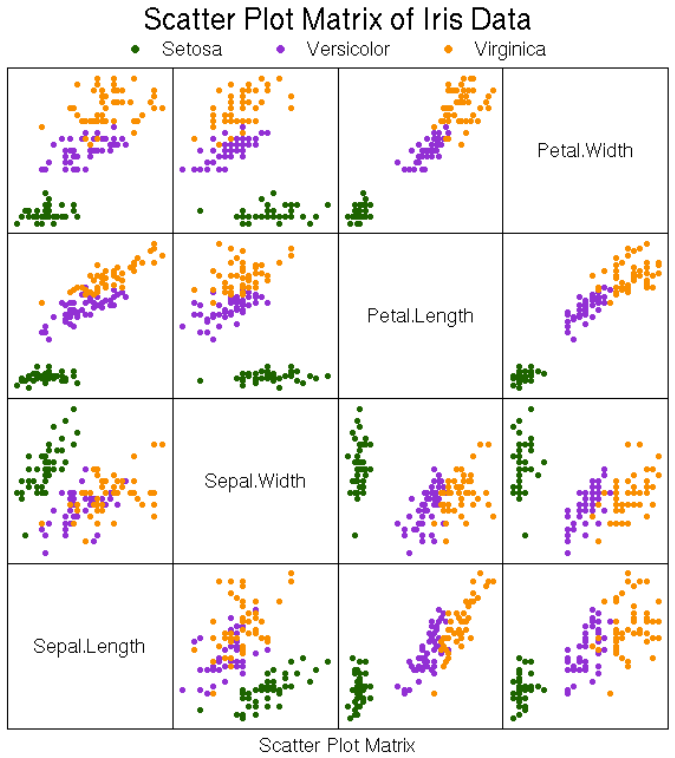
Classification

- Classification is the process of grouping observations (pixels or regions) into classes intended to represent different physical objects or types
- Here, the production of a **thematic map** from (image) data with digital numbers representing for example reflected or emitted EM-radiation in different wavelength bands
- Very many classification methods ranging from quite simple to highly advanced
- Two major groups of methods: supervised and unsupervised
  - supervised: ideally physical classes but not necessarily statistically distinct
  - unsupervised: statistically distinct but not necessarily physical classes

Feature space



- p variables
  C classes
  N observations (or samples)
- $\boldsymbol{x}_i, \ i = 1, \ldots, N, \quad p \times 1$
  is a point (or vector) in p-dimensional **feature space**
- figure shows all possible pairwise projections on original variables



General procedure in supervised classification

- Identify surface types - classes
- Extract training data
- Use training data to estimate model parameters
- Use model to predict the class of all pixels in the image
- Assess the quality of the classification algorithm from test data independent from the training data
- If necessary, refine model by adding more training data

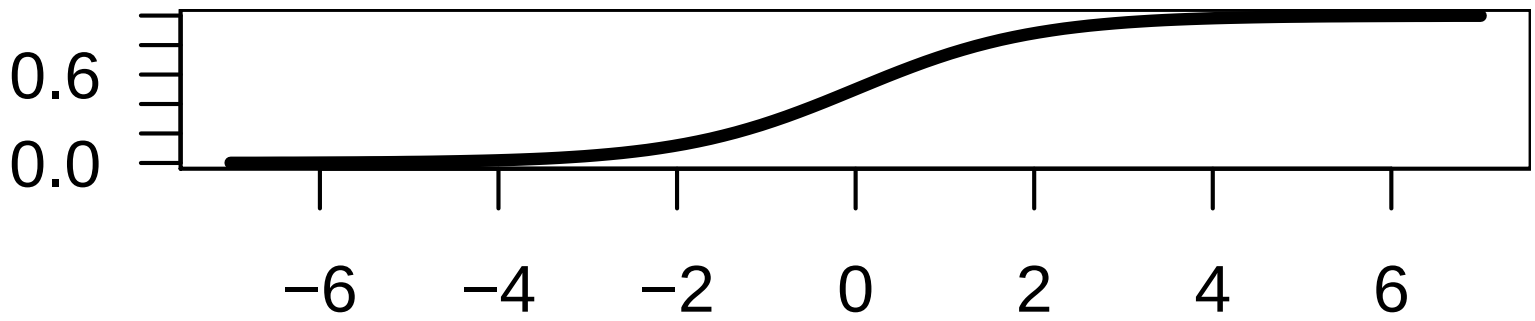
Classification via logistic regression - Binomial distribution reminder

- When we count the number of successes x in n independent experiments with success probability p
- The probability function is

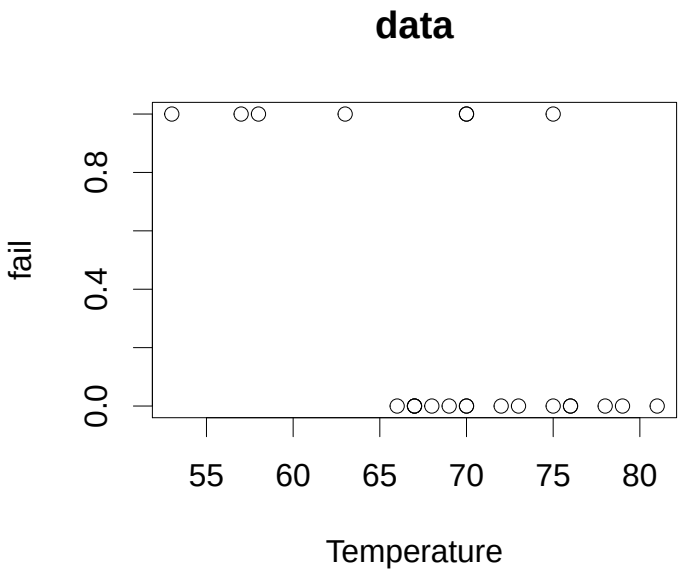
$$p(x, n, p) = \binom{n}{x} p^x (1-p)^{n-x} \quad , \quad x \in \{0, 1, \ldots, n\}$$

- The mean is $E(x) = np$ and the variance $V(x) = np(1-p)$
- The number of experiments n is not a model parameter, but given by the experiment
- The success probability p must be between 0 and 1 so we often parametrize via $\alpha = \text{logit}(p)$, such that:

$$p = \text{logit}^{-1}(\alpha) = \frac{e^\alpha}{1 + e^\alpha}$$



Example of logistic regression: O-ring failures prior to Challenger (1986)



- The data set `oring.dat` contains tests launches of o-rings at different temperatures (in $^\circ$F)
- For each record it is recorded if the o-ring failed.
- The following model is used to describe the data:

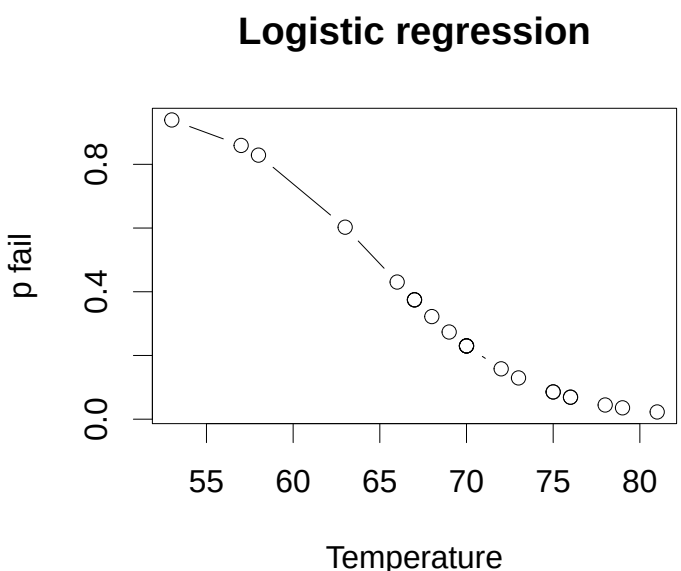
$$fail_i \sim Bin(1, p_i) \ , \quad \text{where logit}(p_i) = \beta T_i + \gamma$$

Example of logistic regression: O-ring failures prior to Challenger (1986)

- On the day of the launch the temperature was 28°F. What was the probability of o-ring failure?

```
dat<-read.table('oring.dat',header=TRUE)
logis.reg <- glm(fail~ temp,family=binomial(link='logit'),data=dat)
pred<-predict(logis.reg)
newdat<-data.frame(temp=28)
pred28<-predict(logis.reg,newdata=newdat)
plogis(pred28)
```

```
##         1
## 0.999805
```



Logistic regression applied as classification

- In the case where we only have/(or want to identify) 2 classes eg. c_1=water and c_2 =rest we can perform a logistic regression to do the classification
- Hence,

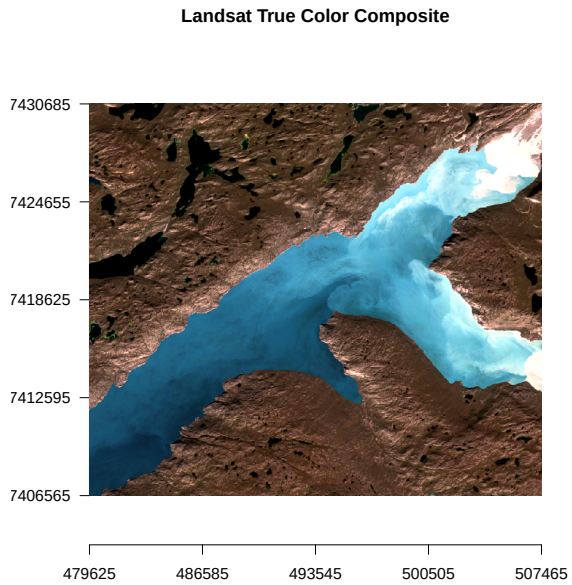
$$X_i \sim \text{Bin}(1, P_i)$$

- The model can be described as

$$\text{logit}(P_i) = \alpha + \beta x_i + \gamma y_i + \dots$$

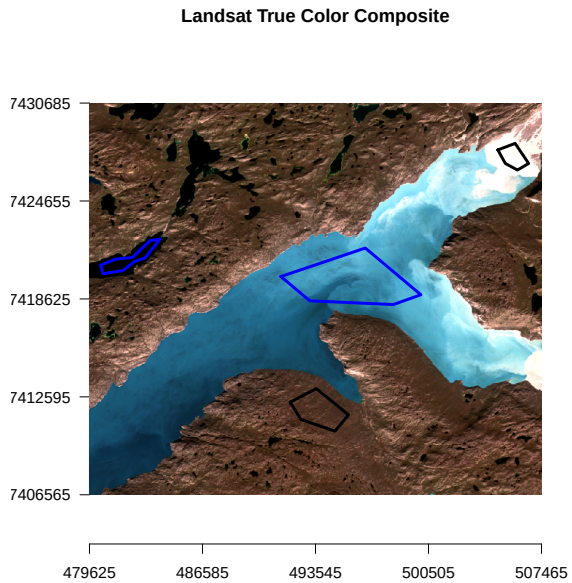
- Here x, y, ... are the bands and $\alpha, \beta, \gamma, \dots$ are the model parameters

Example Landsat image from Greenland, classification via logistic regression



- Landsat image containing 6 bands: blue, green, red, NIR, SWIR1, and SWIR2
- Strategy for the supervised classification
  - Decide on 2 classes e.g. water and rest
  - Select training data
  - Estimate the parameters in the logistic regression based on the training data
  - Calculate the probability of each pixel
  - Assign a pixel to class 1 if the probability is larger than 0.5 and to class 2 otherwise

Select training data



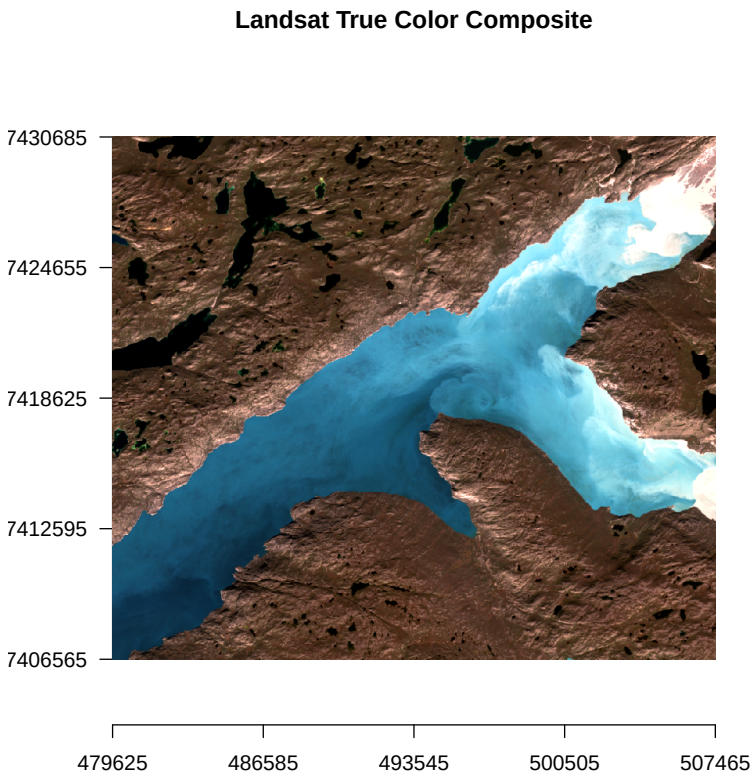
- Here we have selected the 2 classes: "water" and "rest"
- In 'R' we can use the function `drawPoly()` and `extract()` from the 'raster' packages

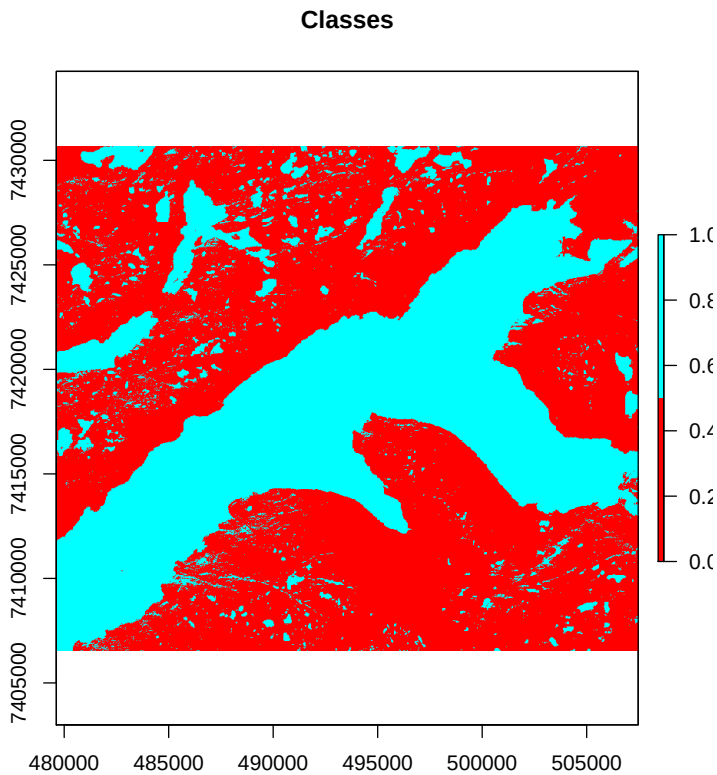
```
nb<-6 # number of features
water<-drawPoly()
#'gr' is the raster image
c.water <- extract(gr, water,cellnumbers=TRUE,nl=nb)
```

- In matlab use `roipoly`
- In python use `cv2.selectROI` from `cv2`

Classification result

```
logis.reg <- glm(V1~ B2+B3+B4+B5+B6+B7,family=binomial(link='logit'),
                 data=trainLogit)
# here V1 is a column in the data set 'trainLogit' containing the class inf
pred<-predict(logis.reg, newdata=gr.df)
cl<-(plogis(pred))>.5
```





Exercise:

Inspired by the previous example

- Use the image Landsat image "yukon.tif" consisting of the bands: blue, green, red, nir, swir1, swir2
- Extract training data
- Create at class variable, so water (lake+ river) is one class and the rest is another class
- First estimate the model parameters based on the training data sets
- Predict the probabilities of each pixel on the logit scale
- Transform the probabilities and assign pixels $P < 0.5$ to class 1 and the rest to class 2
- **Hint** for logistic regression in matlab see here and in python see e.g. here

Probability

- $0 \leq P(A_i) \leq 1$
- $P(\Omega) = 1$
- $P(\cup_i A_i) = \sum_i P(A_i),\ A_i \cap A_j = \emptyset,\ i \neq j$ (disjoint)
- additivity

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

- conditional probability

$$P(A \mid B) = P(A \cap B)/P(B),\ (P(B) > 0)$$

$$P(A \cap B) = P(A \mid B)\ P(B) = P(B \mid A)\ P(A)$$

Bayes Rule

- $A_1, \ldots, A_i, \ldots,$ disjoint and $\sum_i P(A_i) = 1$
- $$P(B) = \sum_i P(B \cap A_i) = \sum_i P(B \mid A_i)\ P(A_i)$$
- Bayes' rule
$$P(A_j \mid B) = P(A_j \cap B)/P(B) = \frac{P(B \mid A_j)\ P(A_j)}{\sum_i P(B \mid A_i)\ P(A_i)}$$
- here
$$P(\omega_c|\boldsymbol{X}) = \frac{P(\boldsymbol{X}|\omega_c)P(\omega_c)}{\sum_{j=1}^{C} P(\boldsymbol{X}|\omega_j)P(\omega_j)} \propto P(\boldsymbol{X}|\omega_c)\ P(\omega_c)$$
- Bayes' classifier: choose maximum $P(\omega_c|\boldsymbol{X})$

Max a posteriori probability MAP

- $P(\omega_c|\boldsymbol{X}) \propto P(\boldsymbol{X}|\omega_c)P(\omega_c)$
- $P(\omega_c)$ is prior (or a priori) probability
- $P(\omega_c|\boldsymbol{X})$ is posterior (or a posteriori) probability
- $P(\boldsymbol{X}|\omega_c)$ is the "likelihood' ', the data term, i.e., the conditional probability of the data given the class
- max a posteriori probability: MAP estimation
- To estimate the MAP, we must assume a probability distribution for our data

Supervised: Gaussian $P(\boldsymbol{X}|\omega_c)$

- Gaussian "likelihood'' in 1-D

$$P(X|\omega_c) = \frac{1}{\sqrt{2\pi}}\frac{1}{\sigma_c}\exp\left[-\frac{1}{2}\left(\frac{X-\mu_c}{\sigma_c}\right)^2\right]$$

  σ_c^2 is variance for class c
- Gaussian "likelihood" in p-D

$$P(\boldsymbol{X}|\omega_c) = (2\pi)^{-p/2}|\boldsymbol{\Sigma}_c|^{-1/2}\exp\left[-\frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c)\right]$$

  $\boldsymbol{\Sigma}_c$ is $p \times p$ dispersion or covariance matrix for class c
  $\boldsymbol{\Sigma}_c$ contains variances on diagonal and covariances off diagonal

Supervised: Gaussian $P(\boldsymbol{X}|\omega_c)$

- MAP: max $P(\omega_c|\boldsymbol{X}) \Leftrightarrow$ max log-likelihood $\mathcal{L}_c(\boldsymbol{X})$ (use Bayes' rule)

$$\begin{aligned}\mathcal{L}_c(\boldsymbol{X}) &= \ln P(\omega_c|\boldsymbol{X}) \\ &= \ln P(\omega_c) + \ln P(\boldsymbol{X}|\omega_c) - \ln \sum_{j=1}^{C} P(\boldsymbol{X}|\omega_j)P(\omega_j)\end{aligned}$$

- The last term on the right hand side is the same for all c, drop it; insert Gaussian $\ln P(\boldsymbol{X}|\omega_c)$

$$\mathcal{L}_c(\boldsymbol{X}) \sim g_c(\boldsymbol{X}) = \ln P(\omega_c) - \frac{p}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma}_c| - \frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c)$$

- drop $-\frac{p}{2}\ln 2\pi$
- if equal priors: drop $\ln P(\omega_c)$

Supervised: Gaussian $P(\boldsymbol{X}|\omega_c)$

- log-likelihood

$$\mathcal{L}_c(\boldsymbol{X}) \sim g_c(\boldsymbol{X}) = \ln P(\omega_c) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_c| - \frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c)$$

  quadratic in $\boldsymbol{X}$: quadratic discriminant analysis

- if equal dispersions, $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$: drop $-\frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}$

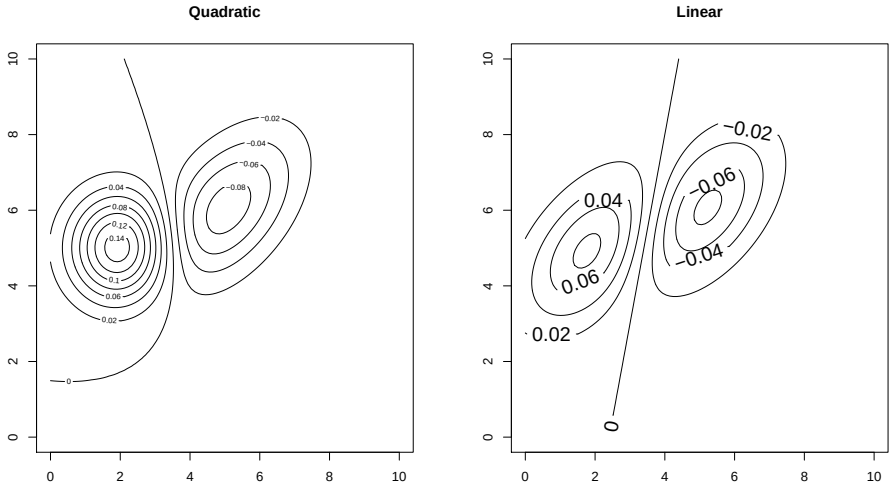
$$\mathcal{L}_c(\boldsymbol{X}) \sim g_c(\boldsymbol{X}) = \ln P(\omega_c) + \boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{X} - \frac{1}{2}\boldsymbol{\mu}_c)$$

  linear in $\boldsymbol{X}$: linear discriminant analysis

- We call $g_c(\boldsymbol{X})$ the discriminant function
- min Mahalanobis distance?
- min Euclidean distance?

Decision surfaces

```
library(mvtnorm)
par(mfrow=c(1,2))
x<-seq(0,10,length=100)
y<-seq(0,10,length=100)
m1<-c(2,5);S1<-diag(2)
m2<-c(5,6);S2<-diag(2)*2; S2[1,2]<-1; S2[2,1]<-1
f<-Vectorize(function(x,y)dmvnorm(c(x,y),m1,S1)-dmvnorm(c(x,y),m2,S2))
g1<-outer(x,y,f)
contour(x,y,g1, main='Quadratic')

m1<-c(2,5);S1<-diag(2)*2; S1[1,2]<-1; S1[2,1]<-1
m2<-c(5,6);S2<-diag(2)*2; S2[1,2]<-1; S2[2,1]<-1
f<-Vectorize(function(x,y)dmvnorm(c(x,y),m1,S1)-dmvnorm(c(x,y),m2,S2))
g2<-outer(x,y,f)
contour(x,y,g2,main="Linear",labcex=1.5)
```



Supervised: Confusion Matrix

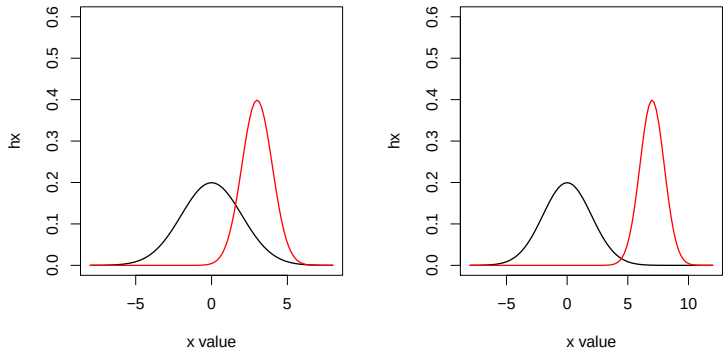
- or error matrix: measures quality of classification result
- resubstitution or training/test

		Classified as									Row total	Producer's accuracy (%)
		bare	urban	tailings	water	forest	tundra	poor veg	waste	wetland		
Known class	bare	6628	1115	0	1	0	1	436	1857	563	10601	62.5
	urban	470	972	1	23	0	1	19	514	286	2286	42.5
	tailings	0	0	1076	0	0	0	0	0	0	1076	100.0
	water	8	17	0	4519	0	0	1	11	26	4582	98.6
	forest	0	0	0	0	1917	176	0	0	0	2093	91.6
	tundra	4	0	0	0	1973	22420	334	0	8	24739	90.6
	poor veg	180	40	0	1	0	91	4801	0	230	5343	89.9
	waste	30	27	0	0	0	0	0	865	29	951	91.0
	wetland	125	371	0	50	0	29	1136	129	6231	8071	77.2
Column total		7445	2542	1077	4594	3890	22718	6727	3376	7373	59742	
Consumer's accuracy (%)		89.0	38.2	99.9	98.4	49.3	98.7	71.4	25.6	84.5		

Threshold probabilities

- In the maximum likelihood classifier we simply assign a pixel to the class where the probability is highest
- However, sometimes all estimated probabilities is on the tail of the given class distribution
- We, therefore might want to define a limit of how small probabilities that are accepted.

$$\boldsymbol{X} \in \omega_c \quad \text{if} \quad g_c(\boldsymbol{X}) > g_j(\boldsymbol{X}) \quad \text{for all} \quad j \neq c \quad \text{and} \quad g_c(\boldsymbol{X}) > T_c$$



- At one of the tails the probabilities of the two classes is similar

$$\ln P(\omega_c) - \frac{p}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma}_c| - \frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c) > T_c$$

Threshold probabilities continued

$$(\boldsymbol{X} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\boldsymbol{X} - \boldsymbol{\mu}_c) < -2T_c - \ln|\boldsymbol{\Sigma}_c| - p \ln 2\pi + \ln p(\omega_c)$$

- The term of the left-hand side follow a χ^2 distribution with p degrees of freedom

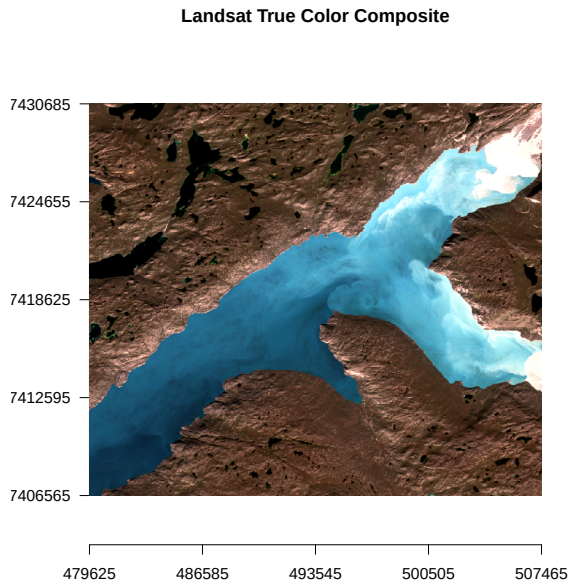
$$T_c = -\frac{1}{2}\chi^2_\alpha - \frac{1}{2}\ln|\boldsymbol{\Sigma}_c| - \left[\frac{1}{2} p \ln 2\pi\right] + \ln p(\omega_c)$$

- If we let $\alpha = 0.95$ and assume we have 6 (bands) degrees of freedom then $\chi^2 = 12.59159$
- The term in the brackets should be included if $g_c(\boldsymbol{X}) = p(\boldsymbol{X}|\omega_c) + \ln p(\omega_c)$

Comments

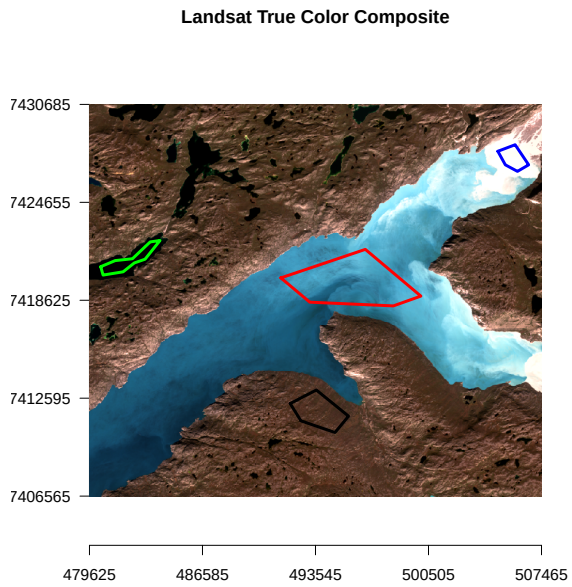
- confusion matrix, learning/test samples, mis classification rate
- histograms of all variables in all classes, derived features (e.g. $\sqrt{X}$, $\ln X$, products, ratios, principal components, local moments, ...)
- calculate posterior $P(\omega_c|\boldsymbol{X})$ for each class
- Quadratic discriminant analysis: p elements in mean vector, $p(p+1)/2$ elements in dispersion matrix, problem for large p
- Simplification
  - regularization
  - diagonal dispersion matrices for each class: p-dimensional Gaussian factors into product of p univariate Gaussians
  - linear discriminant analysis: all classes have same dispersion matrix
  - all classes have dispersion matrix equal to identity matrix
- Mahalanobis distance
- contour curves of constant Mahalanobis distance: for 2-D ellipse (broad, near circle vs thin, elongated), for p-D hyperellipsoid

Example Landsat image from Greenland maximum likelihood classification



- Landsat image containing 6 bands: blue, green, red, NIR, SWIR1, and SWIR2
- Here we assume that the measurement within a class are normal distributed
- Strategy for the supervised classification
  - Decide on the number of information classes
  - Select training data
  - Estimate the parameters μ_c and Σ_c from the training data
  - Calculate the probability of a pixel belonging to a given class
  - Assign a class based on the largest probability

Select training data



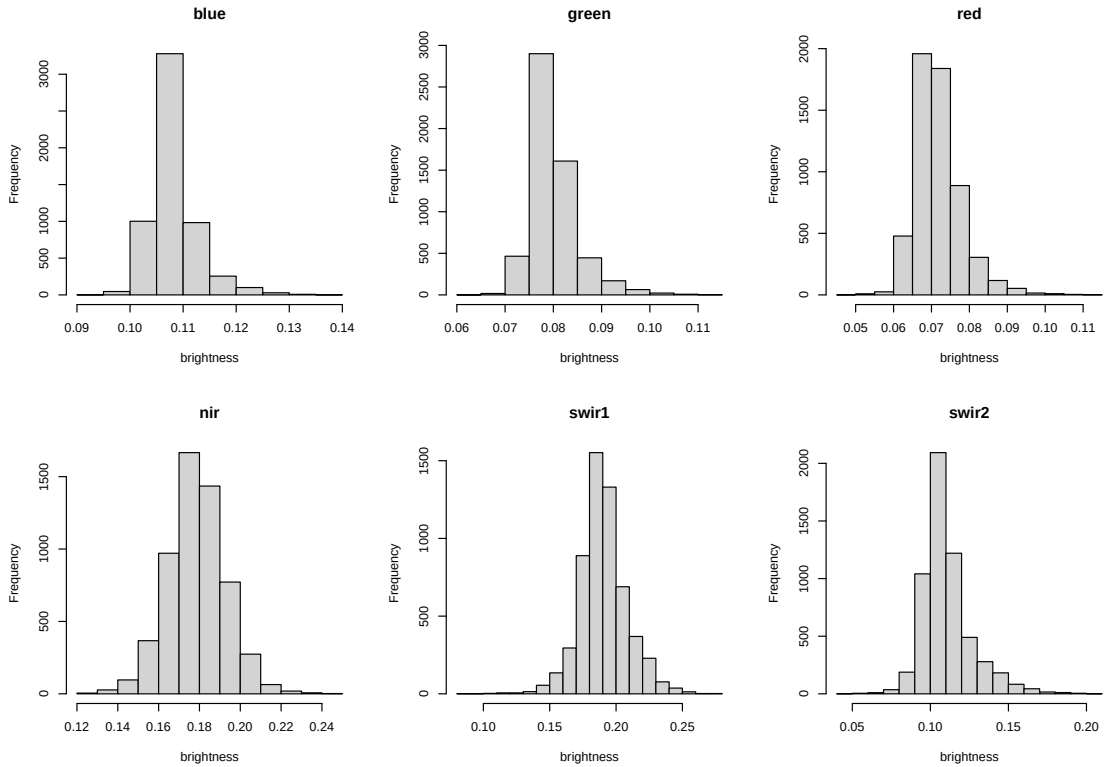
- Here we have selected the 4 classes: “fjord”, “lake”, “rock”, and “ice”
- In ‘R’ we can use the function `drawPoly()` and `extract()` from the ‘raster’ packages

```
nb<-6
rock<-drawPoly()
c.rock <- extract(gr, rock,cellnumbers=TRUE,nl=nb) #gr is the raster image
```

- In matlab use `roipoly`

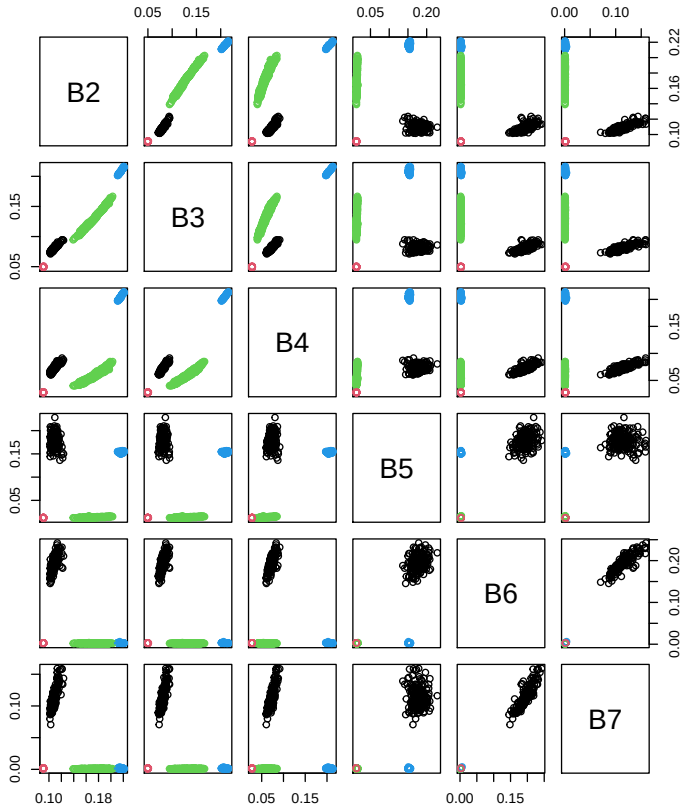
Maximum likelihood classification

- Let us have a look at the training data to see if they can be assumed normal
- Histograms of class 'rock'



- If not we might have to transform

Feature space



- Feature combinations plotted in 2D

Classification

- Hence, we now need to calculate $\log P(X|\omega_c)$
- We assume a normal distribution and that the prior $P(\omega_c) = \frac{1}{4}$

$$\log P(X|\omega_c) = -\frac{p}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{\Sigma}_c| - \frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\mathbf{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c)$$

- This is easily done in R/matlab

```
library(mvtnorm)
dmvnorm(x, mean, sigma, log = TRUE)
```

- Here `x` is the data, `mean` and `sigma` is the mean and covariance estimated from the training data.
- You can also choose the calculate

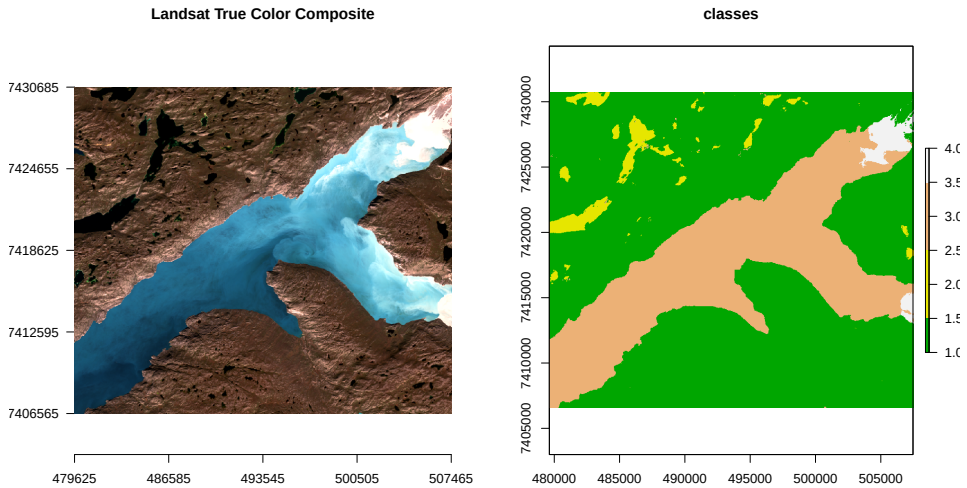
$$\log P(X|\omega_c) \sim -\frac{1}{2}\ln|\mathbf{\Sigma}_c| - \frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_c)^T\mathbf{\Sigma}_c^{-1}(\boldsymbol{X}-\boldsymbol{\mu}_c)$$

- The last part is recognized as the Mahanalobis distance

```
mahalanobis(x, center, cov)
```

Classification result

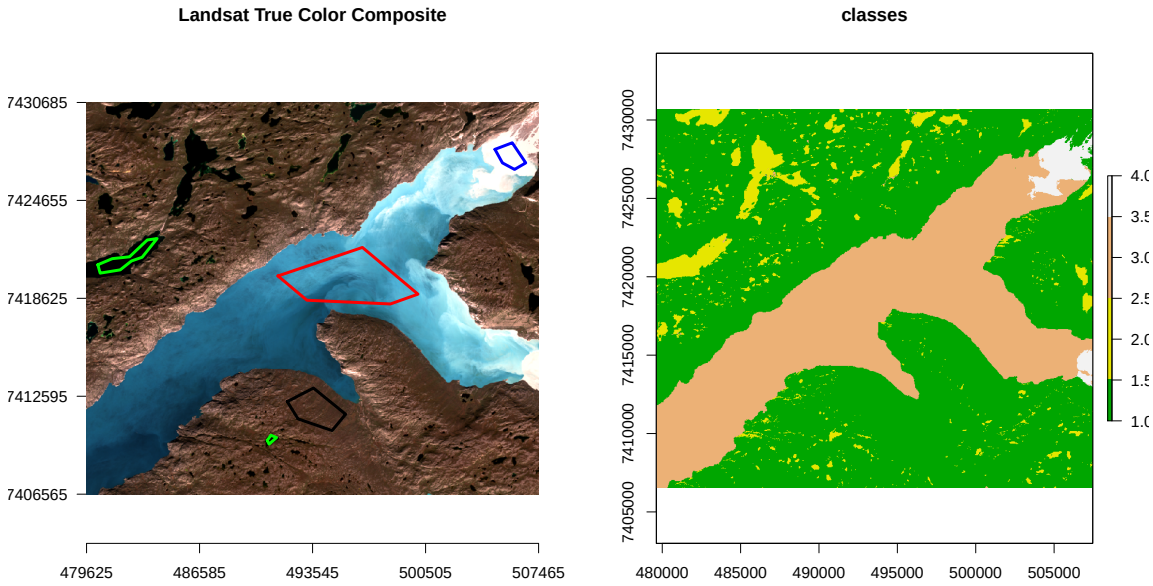
```
nb<-6
load("/home/karina/teaching/ImageRS/scripts/classify/green/train.RData")
mym<-lapply(1:length(train),function(i) apply(train[[i]][,3:(nb+2)], 2, mean))
mycov<-lapply(1:length(train),function(i) cov(train[[i]][,3:(nb+2)]))

library(mvtnorm)
gr.df <- as.data.frame(gr)
logP<-lapply(1:length(train),function(i)dmvnorm(as.matrix(gr.df),mean=mym[[i]],sigma=mycov[[i]],log=TRUE))
logPM<-do.call(cbind,logP)
myclass<-apply(logPM,1,which.max)
gr.class <- myclass
gr$class <- gr.class
```



- The lake class is not well predicted, only the larger lakes are captured
- If the training data does not represent the natural variation, the covariance matrix may not be well estimated
- Choose more training data

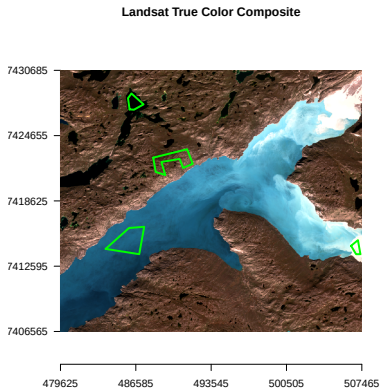
Select more training data and update Classification result

- Let us select more training data and repeat the classification



- The training data for the lake is now more representative
- And the classification result is better, e.g. more lakes are captured

Confusion matrix



- First select test data/ know classes but different from the training data

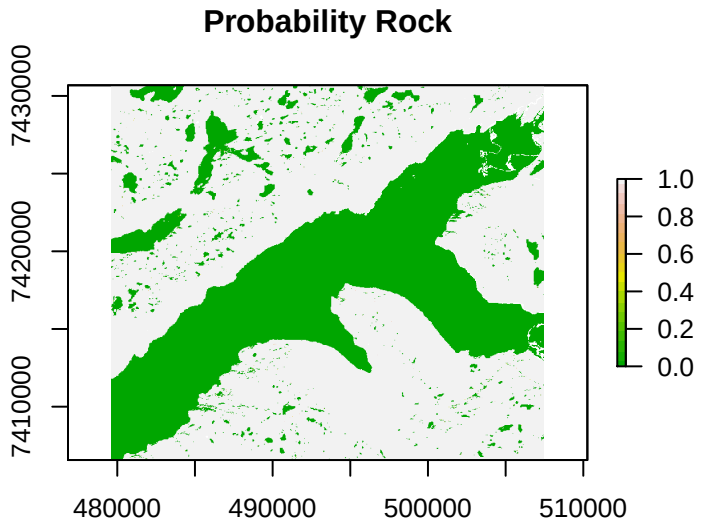
```
#Confusion matrix
confuse<-table(trueClass,predClass)
confuse
```

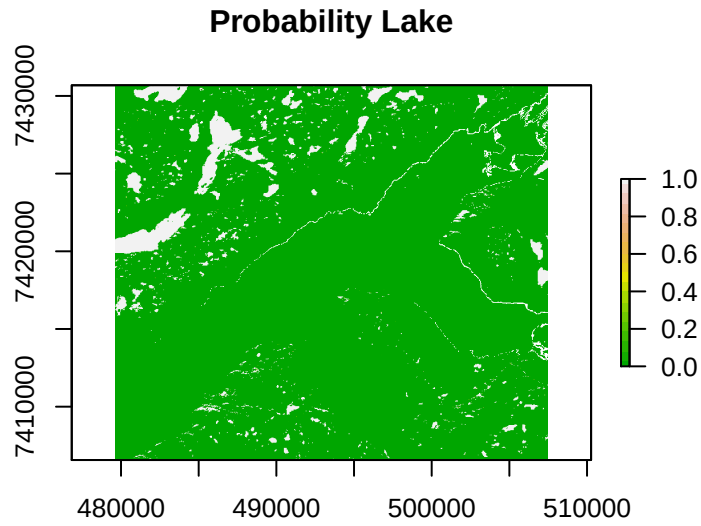
```
##           predClass
## trueClass    1    2    3    4
##         1 4006   38    0    0
##         2    2 1345    3    0
##         3    0    0 5863    0
##         4    0    0  106  588
```

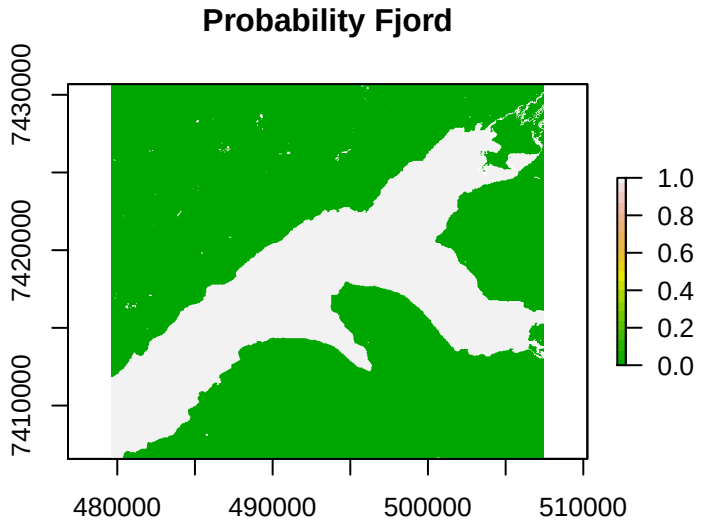
```
# accuracy of classification
sum(diag(confuse))/sum(confuse)
```

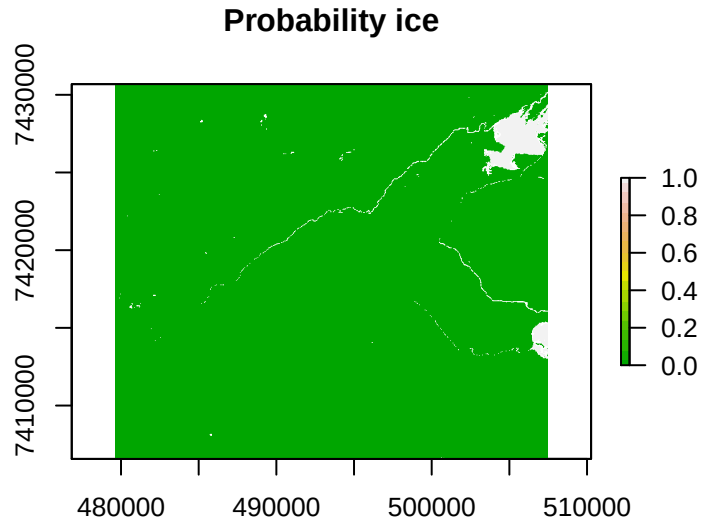
```
## [1] 0.9875324
```

Probability of classes









Other Supervised Methods

- support vector machines, SVM
- tree based methods, CART, random forests
- artificial neural networks, ANN, CNN
- ...

Classification take-away

- Classification is the process of grouping observations (pixels or regions) into classes intended to represent different physical objects or types
- here, the production of a **thematic map** from (image) data with digital numbers representing for example reflected or emitted EM-radiation in different wavelength bands
- very many classification methods ranging from quite simple to highly advanced
- two major groups of methods: supervised and unsupervised

Exercise

- Use the Landsat image 'Yukon.tif' which consists of the bands: blue, green, red, NIR, SWIR1, SWIR2
- Create training and test data sets:
  - Draw polygons on the image to get training data for each class. **Hint** in matlab use the function 'roipoly' in R use 'drawPoly()' in the raster package
  - Extract band values inside the created polygons
- Implement the quadratic discriminant classifier
  - Perform classification based on the original bands
  - Add more training data if needed
- Calculate the confusion matrix based on the test data
- Plot the probability of each class as images
- Additional you can try
  - Add rejection class by setting a threshold
  - Perform classification based on the 3 first principal components
  - Perform a linear discriminant classification
  - Perform the logistic regression on the PCs
  - Compare the water-rest classification with the Normalized Difference Water Index (NDWI)(here you must set et threshold)
  - Perform classification on image from an area of your interest
- **Hint** In matlab the following functions are useful `imread`, `imshow`, `roipoly`, `find`, `reshape`, `mahal`, `max`, `confusionmat`

Report

- Write a small (3-4 pages) report with a brief description of the principles and a presentation of results