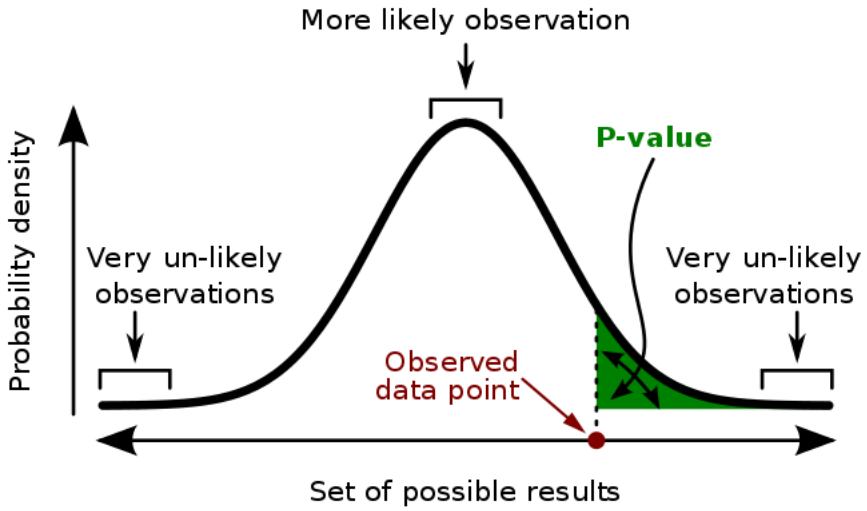
00-Statistical Minimum

p-value

= the propability for a given statistic model, when the null hypothesis is true, that the statistical summary (e.g. the sample mean difference between two groups) would be greater than or equal to the actual observed values

- p-value is the probability of an observed (or more extreme) result assuming that the null hypothesis is true



F-statistics

- F-test measures that two independent random samples are from distribution with same variance == verifies that random samples have similar variance of observed random variable

$$H_0 : \sigma_1^2 = \sigma_2^2 \qquad H_1 : \sigma_1^2 \neq \sigma_2^2 \qquad H_1 : \sigma_1^2 < \sigma_2^2 \qquad H_1 : \sigma_1^2 > \sigma_2^2$$

Multiple comparison problem

- Occurs when one considers a set of statistical inferences simultaneously or infers a subset of parameters selected based on the observed values
- The more inferences are made, the more likely erroneous inferences are to occur

Likelihood function

- function of the parameters of statistical model
- *Probability* describes the plausibility of observed data assumed to be described by a statistical model a parameter value of which is given, without reference to any observed data
- *Likelihood* in this context describes the plausibility of a parameter value of the statistical model assumed to describe the observed data, given specific observed data.

Maximum Likelihood Estimation

- method of estimating parameters of statistical models that maximize the likelihood function using the given observations

- result is called maximum likelihood estimate

01-Dimensionality reduction

Approaches

1. **Linear approaches**
- PCA - and its non-linearization: kernel PCA

2. **Distance preserving approaches**
- multidimensional scaling
- isomap
- locally linear embedding
- tSNE

3. **Self-organizing maps**
- a vector quantization approach
- their relation to k-means clustering

Task definition

Input

$$\mathbf{X} = \{\mathbf{x_i}\}_{i=1}^{m} \subset \mathcal{X} \text{ of dimension } D$$

- assumed: **X** at least approcimately lies on a **manifold** with d < D

Output

- a transformed space T of dimension L
- dimensionality reduction mapping: **F: X -> T**
- reconstruction mapping **f: T->M** ⊂ **X**

such that following holds:
- L < D, L is as small as possible, at best L = d
- the manifold approximately contains all the sample points

$$\{\mathbf{x_i}\}_{i=1}^{m} \underset{\sim}{\subset} \mathcal{M} \overset{def}{=} f(\mathcal{T}),$$

- or alternatively, the reconstruction error of the sample is small

$$E_d(\mathbf{X}) \overset{def}{=} \sum_{i=1}^{m} d(\mathbf{x_i}, \mathbf{f}(\mathbf{F}(\mathbf{x_i})).$$

Manifold Learning

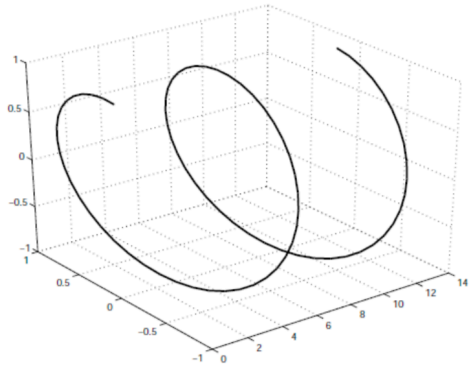
Manifold

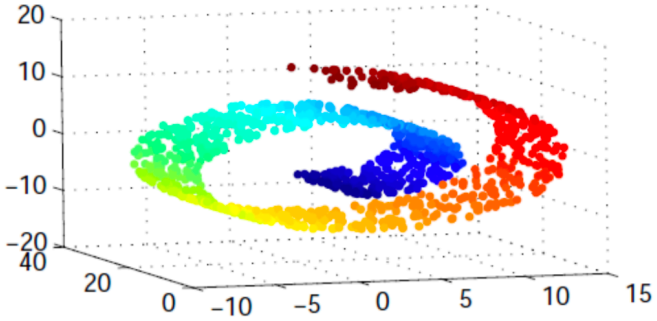
= topological space that on a small enough scale resembles the Euclidean space

- globally tipicaly nonlinear

Learning

- identify a manifold dimension (it is embedded in a space of a higher dimension)
- project the problem(objects) into the low dimensional space - nonlinear dimension reduction
- linear analogy: PCA or factor analysis





Intrinsic dimension

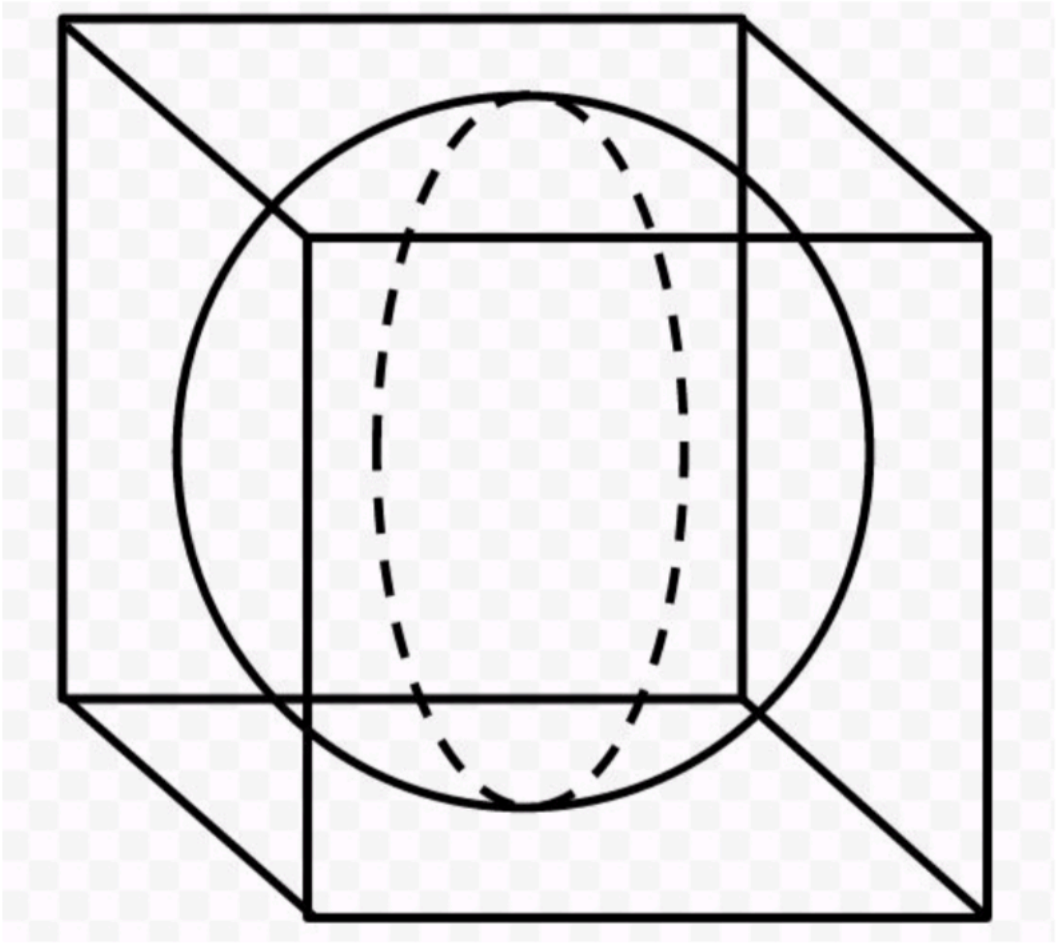
- the number of variables that **v** descibe the phenomenon/data
- when estimated from data, it is a vague number, it depends on:
- the minimum approximation quality criterion or alternatively,
- smoothnes of the manifold

Motivation

- data visualization and understand them
- data compression
- identify hidden causes/latentt variables
- learn in the low-dim space
- possibly obtain better results with fewer training samples in shorter time

The challenges of high-dimensional spaces

- the curse of dimensionality
- in the absence of **simplifying assumptions,** the sample size needed to estimate a function with D variables to a given **degree of accuracy** grows **exponentially** with D
- the geometry of high-dimensional spaces
- empty space phenomenon - ratio of the volumes of unit hypersphere and unit hypercube



PCA

- could be ssen as fitting a ellipsoid to the data
- the new axes have the direction of the highest variance
- they match the aces of the encapsulating/confidence ellipsoid
- in general, PCA **diagonalizes the covarience matrix -** remove linear relationship between variables

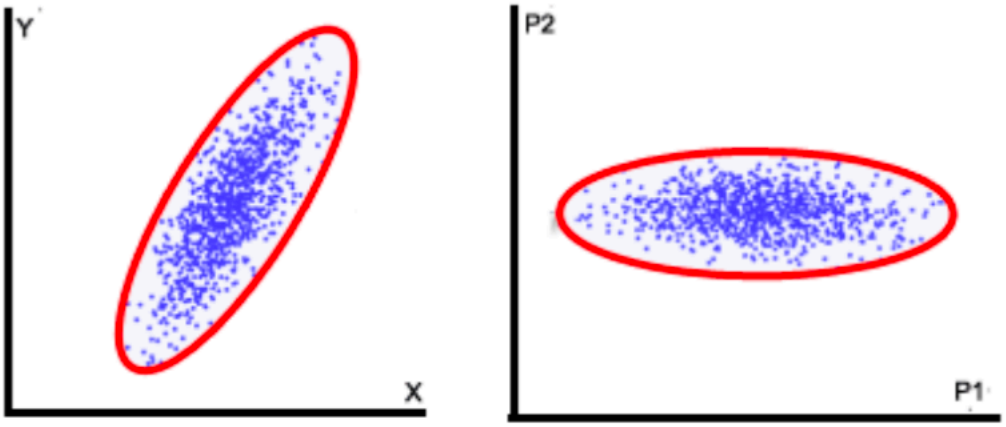
[[σ^2_x},σ^2_{xy}]

[σ^2_{xy},σ^2_y}]]

maximalizuji σ^2_x a σ^2y a minimalizuji σ^2_{xy} (snazim se aby se rovnalo 0)

P1 a P2 (osy krivky PCA) bude linearni kombinaci X a Y

P1 = k1*X+ k2*Y
P2 = k3*X+k4*Y



P1 a P2 (osy krivky PCA) bude linearni kombinaci X a Y

- left image: $\sigma_Y^2 > \sigma_{XY}^2 > \sigma_X^2$,
- right image: $\sigma_{P1}^2 \gg \sigma_{P2}^2$, $\sigma_{P_x P_y}^2 = 0$,

Brief review

For X with zero centered variables

$$\sum_{i=1}^{m} \mathbf{x_i} = 0$$

The covariance matrix can be computed as follows

$$\mathbf{C_X} = \frac{1}{m} \sum_{i=1}^{m} \mathbf{x_i} \mathbf{x_i}^T = \frac{1}{m} \mathbf{X}^T \mathbf{X}$$

by definition PCA constructs a space transformation matrix P_{DxD} such that

$$\mathbf{XP} = \mathbf{T} \text{ and } \mathbf{C_T} \text{ is a diagonal matrix}$$

P is not known, so C_T can't be calculated and diagonalized directly

instead, C_T can be expressed in terms of C_X and P

$$\mathbf{C_T} = \frac{1}{m}\mathbf{T}^T\mathbf{T} = \frac{1}{m}(\mathbf{XP})^T(\mathbf{XP}) =$$
$$= \frac{1}{m}\mathbf{P}^T(\mathbf{X}^T\mathbf{X})\mathbf{P} = \mathbf{P}^T\mathbf{C_X}\mathbf{P}$$

Cx can be decomposed on product of three matrices

$$\mathbf{C_X} = \mathbf{EDE}^T$$

E is the matrix of Cx eigen vectors, D is the diagonal matrix with eigenvalues on diagonal

the only trix is to select P to be a matrix where each column pi is an eigenvector of Cx

$$\mathbf{P} = \mathbf{E}$$

$$\mathbf{P}^T\mathbf{P} = \mathbf{I} \Rightarrow \mathbf{P}^{-1} = \mathbf{P}^T$$

Then it's easy to show that P diagonalizes C_T

$$\mathbf{C_T} = \mathbf{P}^T\mathbf{C_X}\mathbf{P} = \mathbf{P}^T(\mathbf{EDE}^T)\mathbf{P} =$$
$$= (\mathbf{P}^T\mathbf{P})\mathbf{D}(\mathbf{P}^T\mathbf{P}) = (\mathbf{P}^{-1}\mathbf{P})\mathbf{D}(\mathbf{P}^{-1}\mathbf{P}) = \mathbf{D}$$

PCA is solved by **finding the eigenvectors of Cx**

What happens when D is large?

- consider images, the color of each pixel is a feature, megapixel resolution
- large Cx, unfeasible computation of its eigenvectors
- if m is reasonable (m << D) , we can employ following trick
- instead of:

$$\frac{1}{m}\mathbf{X}^T\mathbf{X}\mathbf{u_k} = \lambda_k\mathbf{u_k}$$

we will consider

$$\frac{1}{m}\mathbf{X}\mathbf{X}^T\mathbf{v_k} = \gamma_k\mathbf{v_k}$$

and multiply both sides by X^T

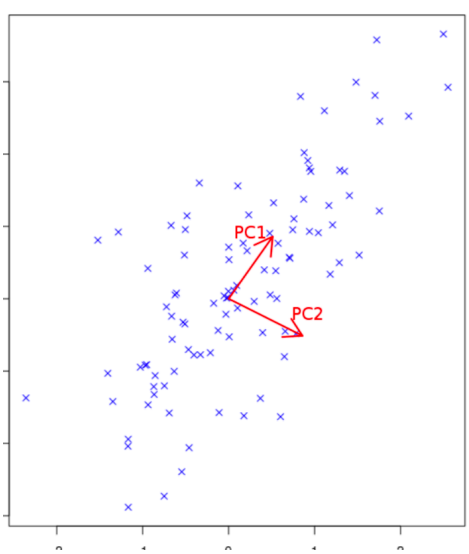
$$\frac{1}{m}\mathbf{X}^T\mathbf{X}\mathbf{X}^T\mathbf{v_k} = \gamma_k\mathbf{X}^T\mathbf{v_k}$$

it's obvious that the substitution matches the original eigenvector decomposition formula

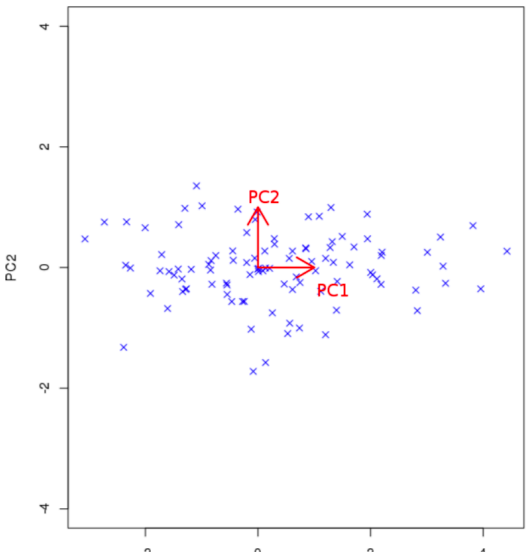
PCA can be solved by decomposition of the m x m scalar-product matrix

3 zpusoby reseni PCA

1. diagonalizace Cx
2. vypocet vl vektoru matice skalarnich soucinu
3. singularni rozklad matice X



$$\mathbf{P} = \begin{bmatrix} \text{PC1} & \text{PC2} \end{bmatrix} = \begin{bmatrix} 0.51 & 0.86 \\ 0.86 & -0.51 \end{bmatrix}$$



Reconstruction mapping - we are multiplying only data with P transponated

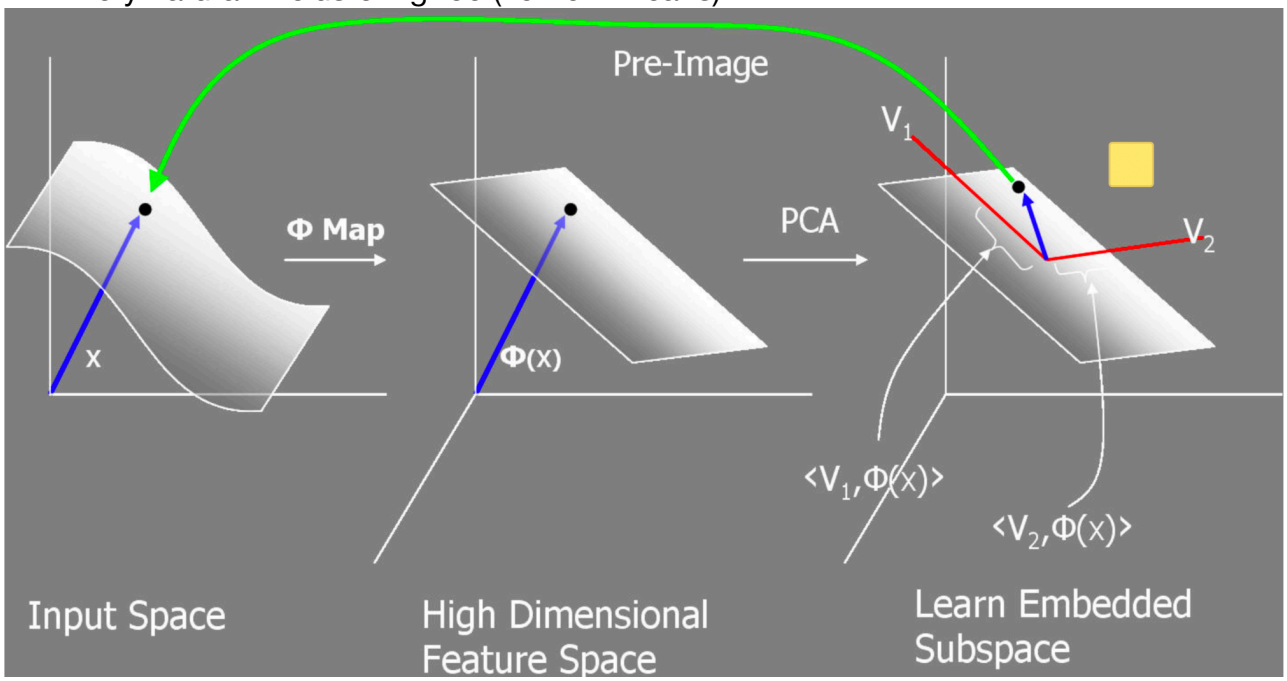
Kernel PCA

- Introduce an intermediate feature space U
- X → U → T - U linearizes the original manifold.
- **feature space transformation** can capture non-linearity

- a domain independent dimensionality reduction, only the transformation tuned for the domain

- **explicit transformation**
- traditional PCA in the new space
- illustrative, but impractical

- **implicit transformation**
- via similarity, resp. kernel function **K:** X x X -> R
- purely a function of object pairs, no object coordinates in the new space
- very natural for clustering, similarity/distance its essential part anyway
- kernel trick analogy (SVM classification)
- an implicit high-dimensional space, classes potentially easily separable
- very natural in clustering too (kernel-kmeans)



- K(xi,xj) = <xi,xj>^2 = <(xi1^2,sqrt(2)*xi1*xi2, xi2^2), …>
- kernel PCA – kernel matrix → diagonalize → a low-dimensional feature space.

Brief review

- We choose a non-linear feature space transformation

$$\phi : \mathcal{X} \rightarrow \mathcal{U}$$

- the transformation is implicit, we only know the kernel function

ition is implicit, we only know the kernel function

$$\forall \mathbf{x_i}, \mathbf{x_j} \in \mathcal{X} : \mathbf{K}(\mathbf{x_i}, \mathbf{x_j}) := \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle = \phi(\mathbf{x_i})^T \phi(\mathbf{x_j})$$

- as with the PCA - we will assume the cov matrix, now in the transformed space

$$\mathbf{C_U} = \frac{1}{m} \sum_{i=1}^{m} \phi(\mathbf{x_i}) \phi(\mathbf{x_i})^T$$

- note that data are assumed to be centered (summing to 0)

$$\sum_{i=1}^{m} \phi(\mathbf{x_i}) = 0$$

- similarly to PCA we will find Cu eigenvectors **v** to decorrelate variables in T

$$\mathbf{C_U v} = \lambda \mathbf{v} \rightarrow \frac{1}{m} \sum_{i=1}^{m} \phi(\mathbf{x_i}) \phi(\mathbf{x_i})^T \mathbf{v} = \lambda \mathbf{v}$$

- ϕ(xi) are not available -> need to replace them by K
- for λ >= 0, v's are in the span of ϕ(xi)
- they can be written as linear combination of the object images

$$\mathbf{v} = \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x_i})$$

- we will substitute for v into the eigenvector formula (two pics above)

$$\lambda \sum_{j=1}^{m} \alpha_j \phi(\mathbf{x_j}) = \frac{1}{m} \sum_{i=1}^{m} \phi(\mathbf{x_i}) \phi(\mathbf{x_i})^T \sum_{j=1}^{m} \alpha_j \phi(\mathbf{x_j})$$

- and use the trick to introduce the dot product, we will multiply by ϕ(xk)T

$$\lambda \sum_{j=1}^{m} \alpha_j \phi(\mathbf{x_k})^T \phi(\mathbf{x_j}) = \frac{1}{m} \sum_{j=1}^{m} \alpha_j \sum_{i=1}^{m} (\phi(\mathbf{x_k})^T \phi(\mathbf{x_i}))(\phi(\mathbf{x_i})^T \phi(\mathbf{x_j}))$$

- the kernel function replaces all the occurrences of ϕ, when iterating over k=1..m we obtain

$$\lambda \mathbf{K} \alpha = \frac{1}{m} \mathbf{K}^2 \alpha$$

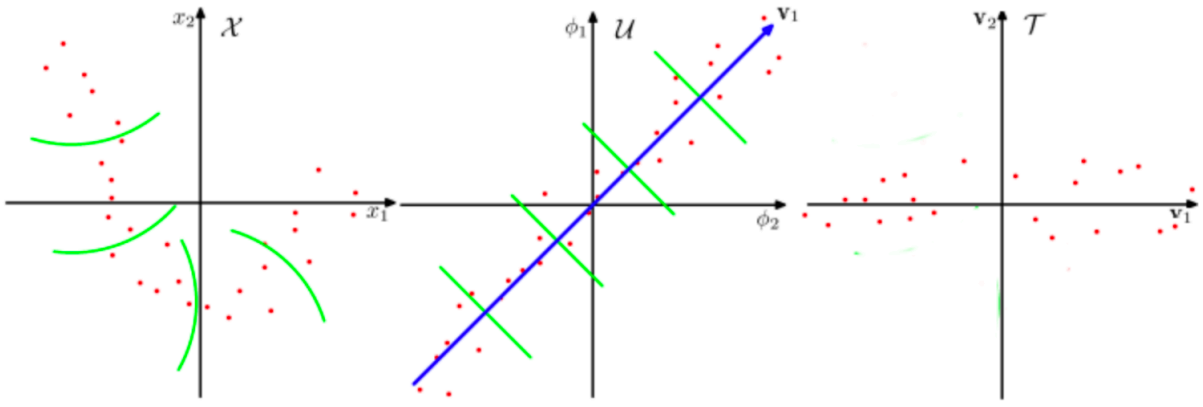
- to diagonalize CT, we solve the eigenvalue problem for the kernel matrix

$$m\lambda\alpha = \mathbf{K}\alpha$$

The last issue is to extract the principal components, the final object images- i.e. the projections of ϕ(xi) onto the eigenvectors in U

$$t_{ik} = \mathbf{v_k}^T \phi(\mathbf{x_i}) = \sum_{j=1}^{m} \alpha_j \phi(\mathbf{x_j})^T \phi(\mathbf{x_i}) = \sum_{j=1}^{m} \alpha_j \mathbf{K}(\mathbf{x_i}, \mathbf{x_j})$$

projekce obrazu objektu xi na vlastni vektory v u



Remarks on kernel PCA

- works for non-linear manifolds
- its complexity does not grow with the dimensionality of U
- one can work with a large number of components too, i.e., increase the dimension
- it can pay-off in subsequent classification
- kernel matrix **K** grows quadratically with the number of data points **m**
- for large data more expensive than PCA
- one may need to subsample the data
- can't get trapped in local minima from the optimization view

- unlike PCA cannot reconstruct object from their principal components - f is unavailable

Multidimensional scaling (MDS)

- the main idea - points close together in X should be mapped close together in T
- minimizes the stress function

$$stress(\mathbf{T}, f) = \sqrt{\frac{\sum_{i,j=1}^{m}(f(\delta_{ij}) - d_{ij})^2}{\sum_{i,j=1}^{m} d_{ij}^2}}$$

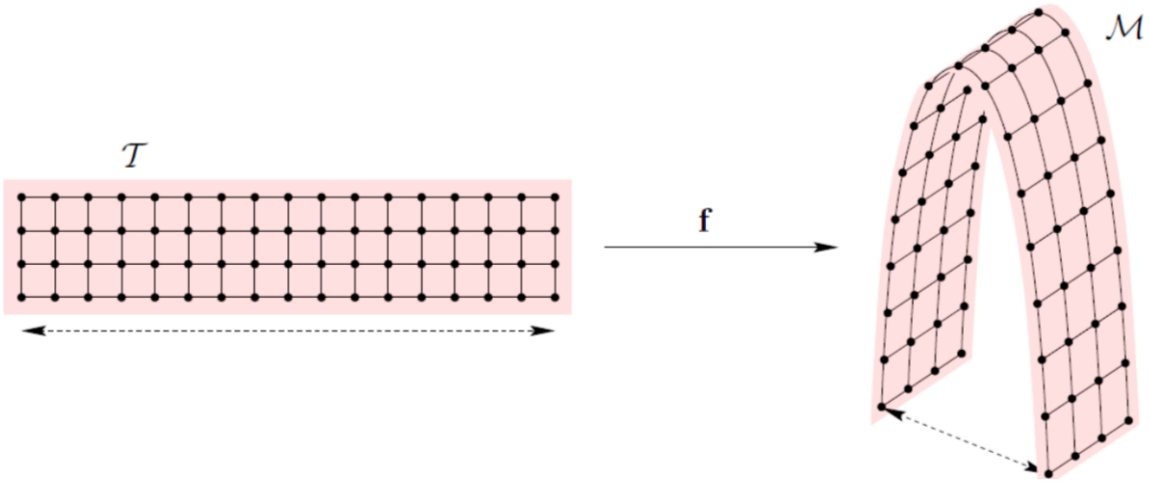
– $\delta_{ij} = d_{\mathcal{X}}(\mathbf{x_i}, \mathbf{x_j})$, $d_{ij} = d_{\mathcal{T}}(\mathbf{t_i}, \mathbf{t_j})$ – typically Euclidean,

– f is a proximity transformation function (e.g., identity, monotonic → metric, ordinal),

- Whole class methods that differs in:
- The method for calculation of proximities delta
- The parametrization of stress function
- The method that minimizes the stress function (e.g.) gradient descent

Geodesic distance

- instead of using non-saying euclidean distance for non-linear manifold we shall better preserve **geodesic distance** betwen points
- a minimum of the length of a path joining both points that is contained in the mainfold
- these patths are called geodesics

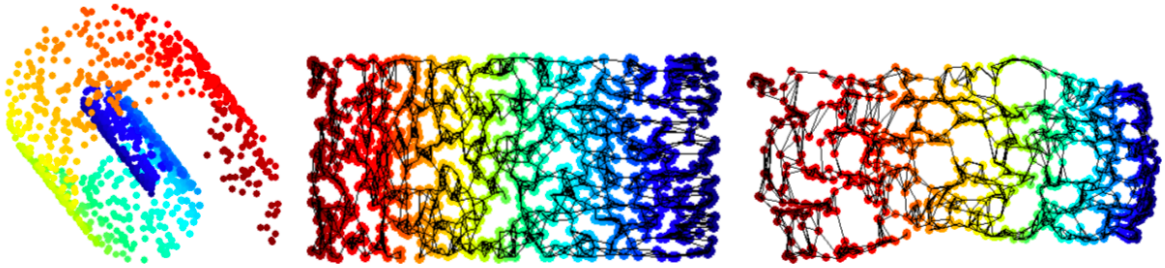


Isomap

- has provable convergence guarantees
- given the infinite input data, the method perfectly recovers the original distance
- cubic complexity, the number of objects can be too much
- finding a proper value of K is not easy

algorithm

1. determine the nearest neighbors
- all points in a fixed radius or K-NN)
2. construct a neigborhood graph
- each points gets connected to its neighbors
- edge length equals the Euclidean dist between points
3. compute the shortest path between all pairs of points
- Floyd, Dijkstra
4. construct a lower dimensional embedding
- use classical MDS



Locally Linear Embedding (LLE)

- Manifold is a topological space that is **locally Euclidean !!**
- the ultimate case of piecewise linear modelling
- approximation of the manifold by a combination of linear models
- a special case of kernel PCA constructing a data-dependent kernel matrix
- for some problems it is difficult to find a kernel for kernel PCA

advantages
- efficient for large datasets - single parameter to tune (K)
- invariant to scaling, rotation and translation

disadvantages
- improper for representing future data
- can be unstable in sparse ares of the input space
- tends to collapse a lot of instances near the origin of T

algorithm
- each data point and its neighbors lie close to a locally linear patch of the maifold
- each point point can be written as a linear combination of its neighbors
- **m** local models, the weights chosen to minimize the reconstruction error

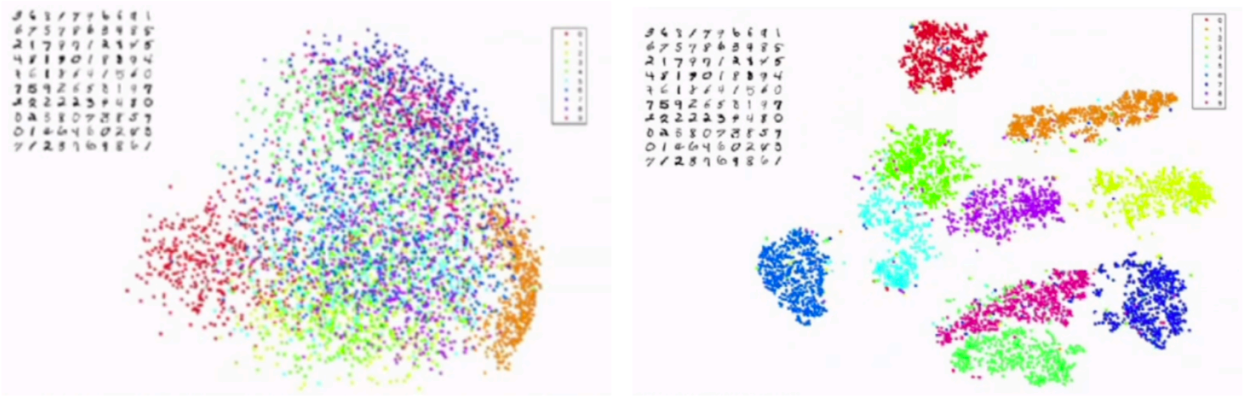
$$\hat{\mathbf{x}}_\mathbf{i} = \sum_{j \in \mathcal{N}(\mathbf{x}_\mathbf{i})} w_{ij}\mathbf{x}_\mathbf{j} \text{ such that } \sum_{i=1}^{m} ||\mathbf{x}_\mathbf{i} - \hat{\mathbf{x}}_\mathbf{i}||^2 \text{ is minimized and } \sum_{j \in \mathcal{N}(\mathbf{x}_\mathbf{i})} w_{ij} = 1$$

- the same weights should reconstruct the point in L dimensions
- global embedding fits the positions t_i in the low-dimensional space

$$\hat{\mathbf{t}}_\mathbf{i} = \sum_{j \in \mathcal{N}(\mathbf{x}_\mathbf{i})} w_{ij}\mathbf{t}_\mathbf{j} \text{ such that } \sum_{i=1}^{m} ||\mathbf{t}_\mathbf{i} - \hat{\mathbf{t}}_\mathbf{i}||^2 \text{ is minimized.}$$

t-Distributed Stochastic Neighbor Embedding (t-SNE)

- distance preserving visualization technique
- puts emphasis on preserving small pairwise distances between objects
- large distances allowed to be modelled as being larger
- as a result, two essential characteristics
- the local data structures retained
- ability to reveal global structure such as the presence of clusters at several scales



PCA tSNE

van der Maaten: Visualizing Data Using t-SNE, GoogletechTalks

Brief review

- the key ides are in the design of the stress function driving gradient descent search
- convert Euclidean distances in both spaces into joint propabilities
- p_ij in the original space X and q_ij in the reduced space T
- minimize their Kullback-Leibler divergence

$$S = KL(P||Q) = \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

The first trick lies in asymmetry of KL divergence

- large p_ij modeled by small q_ij -> **big penalty**
- small p_ij modeled by large q_ij -> **small penalty**
- tends to preserve large p_ij's and thus small distances in the original space

The other tricks consist in definition of p_ij and q_ij

- the empirical propability that an object j is a neighbor of an object i

$$p_{j|i} = \frac{\exp(-||\mathbf{x_i} - \mathbf{x_j}||^2/2\sigma_i)}{\sum_{k \neq i}(-||\mathbf{x_i} - \mathbf{x_k}||^2/2\sigma_i)}$$

- i.e. it is normally distributed wrt their distance
- σi is the kernel bandwidth
- σi is locally adjusted so that a fixed number of objects falls in mode of the Gaussian
- the symmetric joint propability

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2m}$$

In the reduced space T we permit higher probabilities for larges distances

- the normal distribution used in p_ij turns into the heavy-tailed t-distribution in q_ij

$$q_{ij} = \frac{(1 + ||\mathbf{t_i} - \mathbf{t_j}||^2)^{-1}}{\sum_{k \neq l}(1 + ||\mathbf{t_k} - \mathbf{t_l}||^2)^{-1}}$$

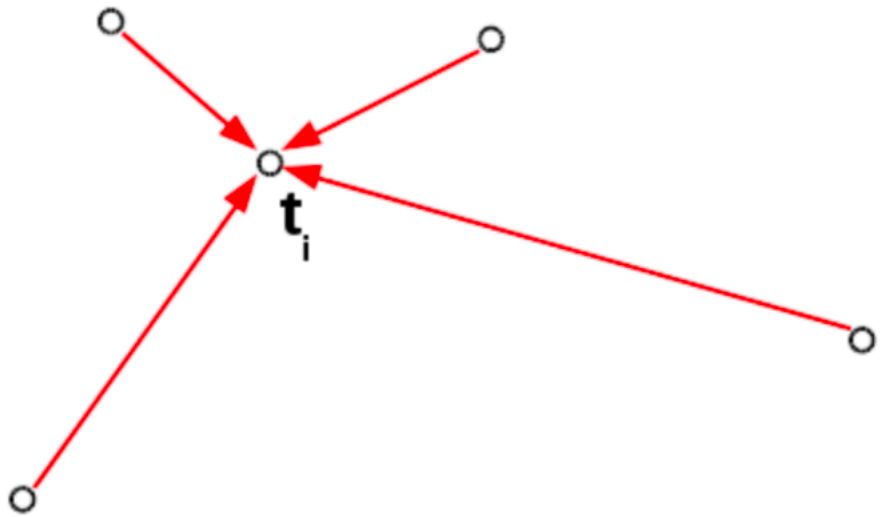
- in KL divergence, p_ij and q_ij shall agree as much as possible, but they may map to differenc distances in both the spaces
- as a result, a moderate distance in the high-dimensional space can be faithfully modeled by a much larger distance in the map
- the reduced map gets insensitive to distant points (they can be placed to many places without big changes in q_ij)

The overall picture

- the gradient descent gradually minimizes the stress function for the individual objects

$$\frac{\partial S}{\partial \mathbf{t_i}} \propto \sum_{j \neq i} (p_{ij} - q_{ij})(1 + ||\mathbf{t_i} - \mathbf{t_j}||^2)^{-1}(\mathbf{t_i} - \mathbf{t_j})$$

- all the other objects get connected via springs that are either stretched or compressed
- the resultant summed force tells us where to move the point in every gradient update



Summary- dimensionality reduction, manifold learning

- Difficult problem namely for the curse of dimensionality
- strong assumptions greatly simplify the task
- the key role of PCA has not been undermined by any non-linear method yet
- they work for well-sampled smooth manifolds, but not necessarily for real data
- besides the curse of dimensionality, the problems could be caused by insufficiency of objective functions or numerical problems durin optimization
- there is a large pool of non-linear reduction methods
- the key properties are effectivity and efficiency including convergence

02-Clustering

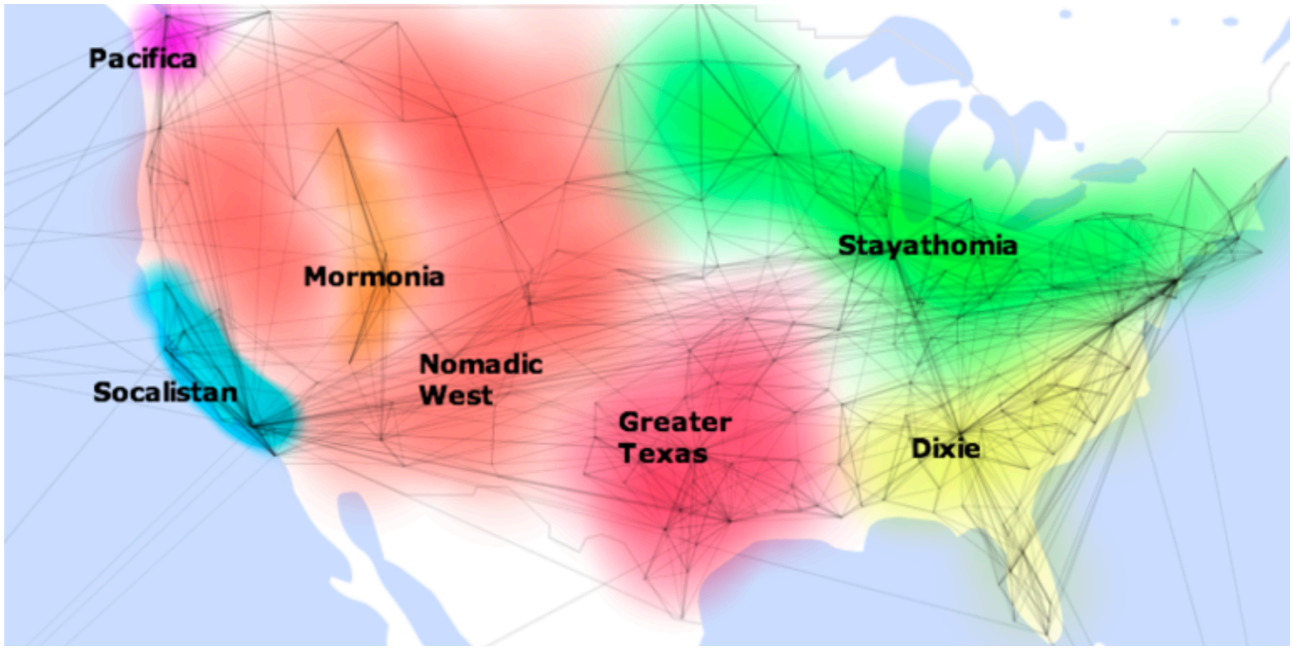
Application

- image segmentation - features - (a) color components, (b) brightness for b&w images

- clustering for learning
- class discovery in (unannotated) data
- unsupervised learning

- data understanding
- outlier detection

- usage of prototypes
- summarization
- compression
- eficient nn search



Formalization

Goal

- split unclassified object into mutually disjoint subsets - **clusters**
- we divide so that the objects

1. are similar inside cluster
2. are dissimilar when lying in different clusters

we solve an **optimization problem**

- inputs
- training data
- distance function
- (optimization criterion
- unknown

- the number of clusters
- partition - cluster-object links

Complexity

- variant of Bayesian Decision Task

how large space to be searched?

- the optimization criterion cannot be applied in a naıve way (exhaustive search) -> NP-hard problem
- need to come up with heuristic solutions

K-Means

- greeedy algorithm
- guaranteed convergence, typically fast
- finds a locally optimal solution
- sensitive to initialization -> can stuck easily in local optima

- global homegeneity criterion

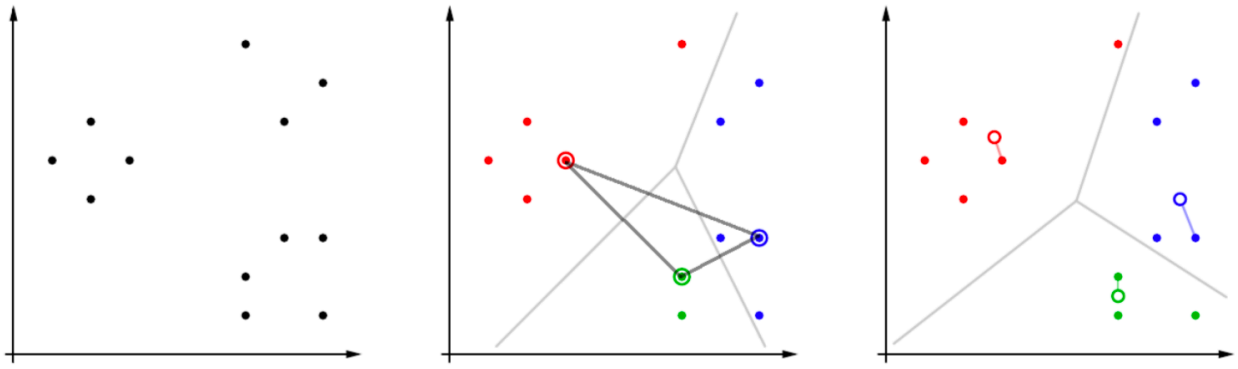
$$W(k) = \underset{\Omega}{\operatorname{argmin}} \sum_{i=1}^{k} \sum_{x_j \in C_i} d(x_j, \mu_i),$$

Distance function

- typically **metric** on X
- common functions:
- Minkowski metric - selection of k (manhattan, euclid, chebyshev)

$$d(x, y) = \left(\sum_{i=1}^{n} (x_i - y_i)^k \right)^{\frac{1}{k}}$$

- Levenshtein distance (words, strings)



Expectation Maximization (EM) algorithm

- kmeans is an EM algorithm specialization
- maximizes **likelihood** P(X|Θ)

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, Pr(\mathcal{X}|\theta) = \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^{m} Pr(x_i|\theta)$$

- introduces a auxiliary variable alpha, which simplifies maximization of P(X|θ)

E-step

- estimate alpha variables w.r.t. current parameter values θ

M-step

- modify parameters θ so that likelihood is maximized wrt current alpha

K-Means specification

- alpha gives binary cluster membership
- E-step: assign objects and centroids
- M- step: recalculate cluster centroid

EM clustering - K-Means Comparision

- clustering defined as GM optimization in n dimensions
- the number of elements- k
- partition: object belongs to distribution with highest posteriori prob P(C|x)
- assumes a normaln object distribution within a cluster
- more robust, but slower than K-Means

Soft (probabilistic) clustering

- hard object membership in a single cluster not needed
- soft clustering is a special case of **fuzzy clustering**
- membership function P(Cj|xi) is understood as probability

$$-\ \text{it must hold: } \forall i = 1, \ldots, m : \sum_{j=1,\ldots,k} Pr(C_j|x_i) = 1$$

- Soft clustering algorithm- “soft” k-means
- EM principle
- a model with parameters θ ysed ti calculate P(Cj|xi)
- θ most often defines as **Gaussian Mixture Model - GMM**

EM GMM Clustering

- Q determines probability that an object was generated by a particular distribution

Soft clustering with a naive bayes (NB) model

- nb classifier, samples with known classes

$$Pr(C_j|X_1 = v_1, \ldots, X_n = v_n) = \frac{Pr(C_j) \prod_{i=1}^{n} Pr(X_i = v_i|C_j)}{Pr(X_1 = v_1, \ldots, X_n = v_n)}$$

EM when classes are not available:

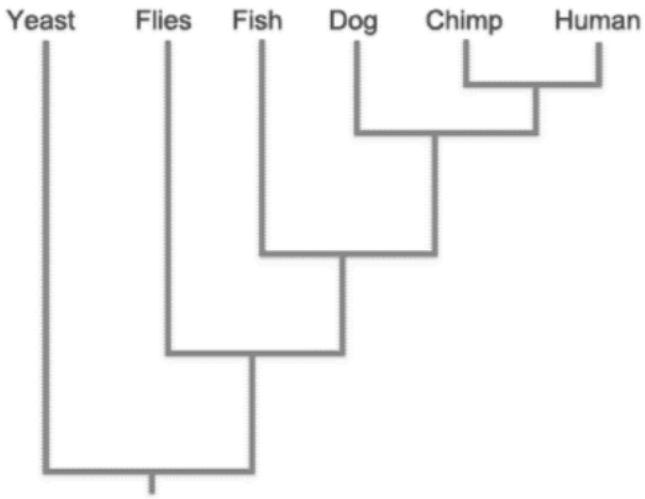
1. initialize: augment the data with the class count column (randomly, class priors)

2. M-step: infer the model from the augmented data, use MLE
3. E-step: update the augmented data based on the model, use Bayes formula
4. repeat 2.3. until the changes are samll enough

Hierarchical clustering

Motivation

- taxonomy is more informative than partition
- analyzes on various granularity levels
- binary tree = dendogram
- a reasonable decomposition of the clustering problem to subproblems
- a straightforward and computationally efficient solution



Algorithm

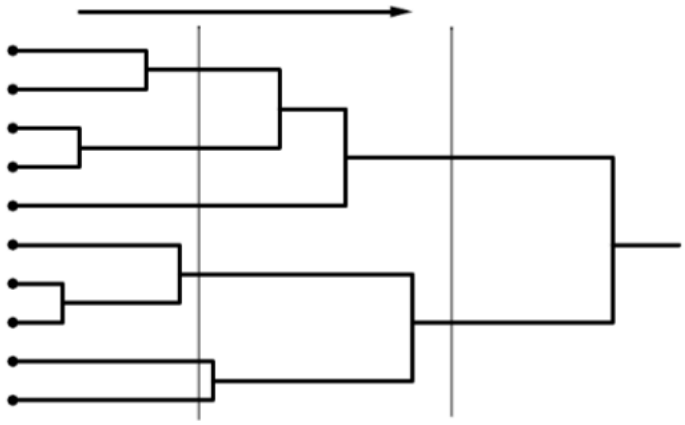
- recursive application of the standard clustering step
- needs no prior k, constructs a hierarchy
- a partition results from dendogram cut

agglomerative approach (bottom up)

- at the beginning each object makes a cluster
- iterate with merging the most similar clusters, typically pairs

divise-approach (top-down)

- split the object set into cluster, typically two of them
- iterate with splitting the clusters
- more difficult to implement - needs an internal clustering algorithm
- more efficient than agglomerative, namely when the complete dendogram not needed



Summary (Clustering)

- intuitively comprehensible principle, in many context, in many domain
- in general identiification of any frequent event co-occure in data
- combinatorially difficult optimization problem
- heuristic solutions, local optimality
- basic steps
- representation definition
- distance function selection
- clustering itself
- abstract representtion of partition
- evaluation, iteration

clustering algorithm quality
- scalability - no of objects, dimensions
- robustness - noise, outliers, feature types, distance function
- ability to deal with various cluster shapes

Method categorization

Nonhierarchival methods
- aim to deliver the partition that minimizes an optimization criterion
- apply a global homogeneity criterion
- cluster membership can be hard (K-Means) as well as probabilistic (GMM)

Hierarchical methods
- generate a cluster hierarchy
- binary tree = dendogram

- apply a local cluster similarity criterion
- agglomerative - bottom-up
- divise - top-down, divide and conquer
- examples: AHC(general principle)

03-Manova

Dependent vs independent variables

- independent variable is the variable that being changed or controlled in a scientific experiment to test the effect on the dependent variable
- dependent variable is the variable that is being tested and measured in a scientific experiment
- the dependent variable is `dependent` on independent variable. As the experimenter changes the independent variable, the effect on dependent variable is observed and recorded

Bivariate statistical models and tests

- asses strength of relationship between a pair of variables
- independent (causal) and dependent (effect) variable
- rejection of null hypothesis does not imply causal relationship
- all of them can be generalized towards multivariate statistics

		dependent variable	
		categorical	continuous
independent variable	categorical	contingency table chi-square test	analysis of variance
	continuous	LDA logistic regression	correlation regression

Independence test for two categorical variables

- categorical variable
- takes one of a limited (and fixed) number of possible values
- contingency table
- table showing observed (multivariate) joint frequency distribution
- for the moment concern two-way contingency tables only
- pair of variables with **r** and **c** categories captured in a **(r x c)** table
- its elements represent frequency counts for the individual events

	healthy	diseased	total
women	216	72	288
men	279	342	621
total	495	414	909

	X_{21}	$\dots$	X_{2c}	Σ
X_{11}	N_{11}		N_{1c}	$N_{1\bullet}$
$\dots$				
X_{1r}	N_{r1}		N_{rc}	$N_{r\bullet}$
Σ	$N_{\bullet 1}$		$N_{\bullet 2}$	N

- independence assumption
- H_0: two categorical variables are independent
- H_a: they have an association or relationship (some)
- the frequency distribution does not change with the table rows

- compare the observed frequencies with the expected ones
- the expectations are derived from the marginal frequencies under the independece assumption, MLE approach is taken
- let us measure the discrepancy between the observed counts and the estimated expected counts under the null

Pearson's chi square

- one of the options

$$X^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

- a cumulative test statistic
- it asymptoticaly approaches a chi-square distribution
- with (r-1)(c-1) degrees of freedom

Assumptions

- non-parametric test, robust wrt distribution of the data
- one observation per subject, sufficient sample size (E_ij >= 5)

Example

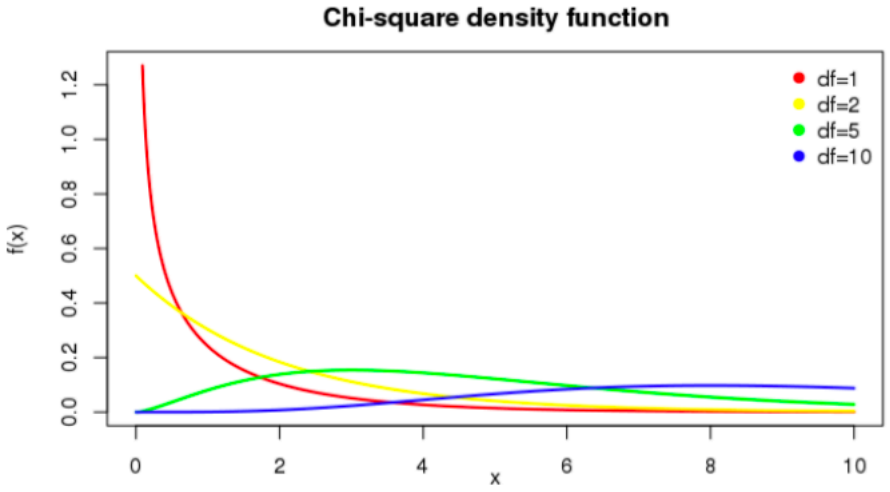
- gender and disease relationships

$$X^2 = \frac{(216 - 157)^2}{157} + \frac{(72 - 131)^2}{131} + \frac{(279 - 338)^2}{338} + \frac{(342 - 283)^2}{283} = 71.3$$

- choose a significancy level α=0.01
- compare with the table value

$$\chi^2_{\alpha=0.01,df=1}=6.635,$$

- since X^2 > chi-square_df=1 : reject H_0
- the corresponding p-value: p =1 - F_{chi-square(1)}(71.3) = 1.09e-17



Categorical dependent vs continuous independent variable

- review **t-test for two groups**
- a test in which the test statistics follows a Student's t-distribution
- under the null hypothesis

- consider a two sample t-test

$$H_0 : \mu_1 = \mu_2,\ H_a : \mu_1 \neq \mu_2$$

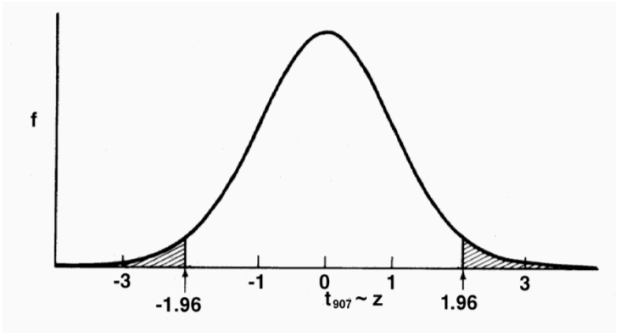
- the two populations should follow a normal distribution
- variances of the two populations assumed equal -> **Student's t-test**
- variances can differ -> **Welch's test**

$$t_{obs} = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad \sim \quad t_{df}$$

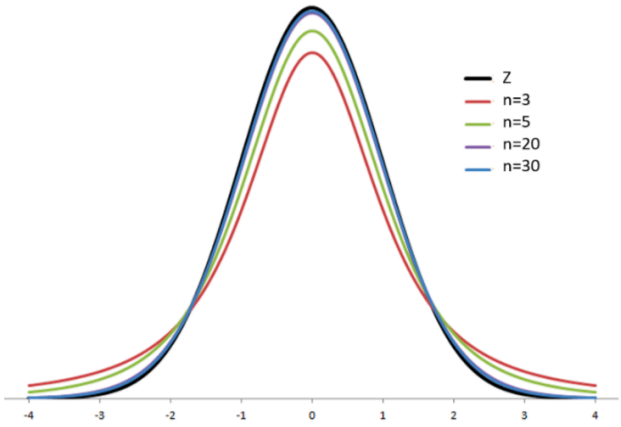
X~_i, s_i^2, n_i … sample means, variances and sizes

- df <= n_1 + n_2 -2, the exact formula complicated
- reject H_0 if:

$$|t_{obs}| \geq t_{df,1-\alpha/2}.$$



Statlect: The Digital Textbook



Statlect: The Digital Textbook

T-test for multiple groups

- Concern a categorical variable with many levels -> multiple groups

$* \ H_0 : \mu_1 = \mu_2 = \cdots = \mu_g,$

$* \ H_a : \mu_i \neq \mu_j$ for at least one $i \neq j$.

- conduct a two-sample t-test for a difference in means for each pair of groups
- the number of comparisons grows quadratically with the number of groups.levels
- for α=0.05 for each comparison
- there is a 5% chance that each comparison will falsely be called significant
- we falsely reject at least one of the partial null hypothesis with probability

$$1 - (1 - \alpha)^{\binom{g}{2}}$$

e.g. for g=4 it makes 0.26 >> α

multiple comparisons must be corrected!

often we control **family-wise error rate (FWER)**

- the probability of making one or more false discoveries
- the most simple FWER control is the **Bonferroni correction**
- test each hypothesis at level α_indiv = α_overall / m
- m stands for number of individual pair test
- follows from Bonferroni inequality for independent tests

$$\alpha_{overall} = 1 - (1 - \alpha)^m \leq m\alpha_{indiv}$$

- in our case with 4 groups m (4 over 2) = 6
- the Bonferrony inequality obviously holds
- 0.26 = 1 - 0.95^6 < 0.05 * 6 = 0.3
- this adjustment may be too **conservative**
- insufficient **power**, often doesn't reject H_0 although H_a is true

Analysis of variance (ANOVA)

- $H_0 : \mu_1 = \mu_2 = \cdots = \mu_g,$
- $H_a : \mu_i \neq \mu_j$ for at least one $i \neq j.$

- compares means for multiple (g>3) independent populations
- parametric and unpaired, one-way
- relationship between a categorical factor F and a continous outcome Y
- extends a two sample t-test for mulitple groups

Subject	F	Y
1	f_1	y_1
2	f_2	y_2
...		
N	f_N	y_N

		1	...	g
Subject	1	y_{11}	...	y_{g1}
	2	y_{12}	...	y_{g2}
	...	...	...	...
	n_i	y_{1n_1}	...	y_{gn_g}

y_ij … observation for subject j in group i
n_i … number of subjects in group i
N = n_1 + n_2 + …+ n_g ... total sample size

Assumptions

- the subject are **independently sampled**
- employ repeated measures ANOVA otherwise
- the data are **normally distributed**
- E(Y_i) = μ_i e.g. no group sub-populations with different means
- employ non-parametrix Kruskal-Wallis tesst otherwise
- the data are **homoscedastic**
- the variability in the data does not depend on group membership
- there is a common variance var(Y_{ij}) = σ^2

Method

- partition **SS_total**, the total variation in a response variable
- distinguish **within groups variability SS_error** and **between groups variability SS_treat**

$$\underbrace{\sum_{i=1}^{g}\sum_{j=1}^{n_i}(y_{ij}-\bar{y}_{i.})^2}_{SS_{error}}+\underbrace{\sum_{i=1}^{g}n_i(\bar{y}_{i.}-\bar{y}_{..})^2}_{SS_{treat}}$$

- in a similar manner, partition the number of degrees of freedom that stand behind the observed sums of the squared deviations

$$DF_{total}=N-1=DF_{error}+DF_{treat}=(N-g)+(g-1)=N-1$$

- decide whether group averages differ more than based on random variability observed in the dependent variable under the null hypothesis
- employ **mean square** variability, both within group and between groups

$$MS_{error}=\frac{SS_{error}}{DF_{error}}=\frac{SS_{error}}{N-g} \qquad MS_{treat}=\frac{SS_{treat}}{DF_{treat}}=\frac{SS_{treat}}{g-1}$$

- ANOVA compare the variance between the groups and within the groups

$$F_{obs}=\frac{MS_{treat}}{MS_{error}}\sim F_{g-1,N-g}$$

F_obs is small (close to 1)
- variability between groups is negligible compared to variation within groups
- the grouping does not explain much variation in the data

F_obs is large
- variability between groups is large compared to variance within groups
- the grouping explains a lot of the variation in the data

Decision rule based on F_obs

- reject H_0 if $F_{obs}\geq F_{\alpha,g-1,N-g}$,
- fail to reject H_0 if $F_{obs}<F_{\alpha,g-1,N-g}$.

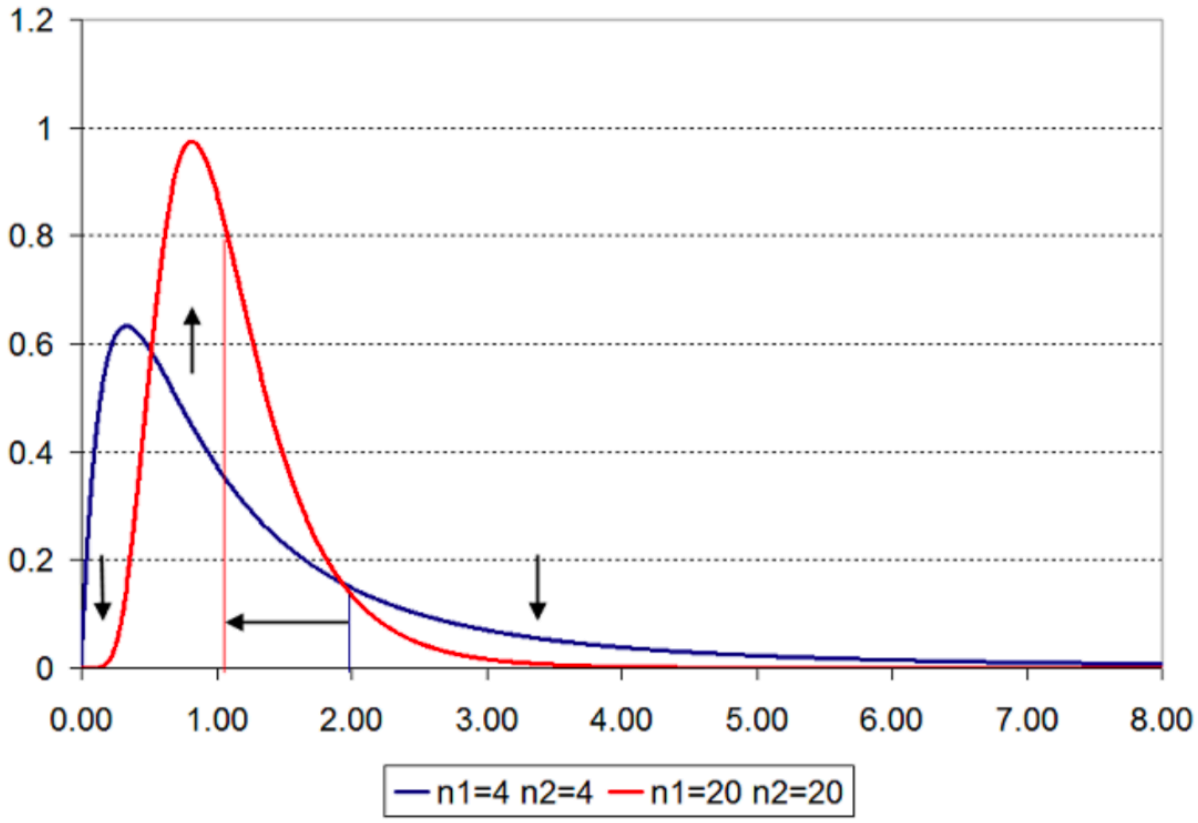
F-distribution

- any distribution obtained by taking the quotient of two chi-square distributions divided by their respective degrees of freedom

- consequently, any F-distribution has two parameters corresponding to the degrees of freedom for the two chi-square distributions
- given:

$$X_1 \sim \chi^2_{df_1} \text{ and } X_2 \sim \chi^2_{df_2}$$

$$\frac{X_1/df_1}{X_2/df_2} \sim F_{df_1,df_2}$$



Post-Hoc Anova tests

- after performing ANOVA (and rejecting H_0)
- we only assume that there is some difference in group means
- a post-hoc test identifies which particular groups stand behind the test outcome

Tukey's HSD

- **h**onest **s**ignificant **d**ifference test
- compares all pairs of group means
- identifies all pairwise difference is larger than expected standard error
- observed test statistics related to the studentized range distribution

$$q_{obs} = \frac{\bar{y}_{i.} - \bar{y}_{j.}}{\sqrt{\frac{MS_{error}}{n^*}}} \quad \sim \quad q_{g,N-g}$$

n*…number of observation per group
- q obs is computed for each pair of groups
- q obs increases when the mean between groups is different

ANOVA extensions

ANOVA

- is parametric
- deals with independent measurements
- is one-way
- concerns a single target variable only

Other options

- non-parametric analysis (Wilcoxon test -> Kruskal-Wallis analysis)
- compares all possible group means (repeated measures ANOVA, Friedman test if non-parametric too)
- main effects ANOVA and factorial ANOVA
- multivariate ANOVA - **MANOVA**

Multivariate Analysis of Variance (MANOVA)

- **p** variables measured on each subject, object categorized into g disjoint groups
- y_ijk…an observation for variable k from subject j in group i
- y_ij… a vector of dependent variables for subject j in group i

- $H_0 : \mu_1 = \mu_2 = \cdots = \mu_g,$
- $H_a : \mu_{ik} \neq \mu_{jk}$ for at least one $i \neq j$ and at least one variable k.

Assumptions

- subjects are **independently sampled**
- the dara are **multivariate normally distributed** in each group
- the data from all groups have common covariance matrix ∑
- the data from group i has common mean vector μ_i of length p

Method

- the analogy of SS_total in ANOVA is (p x p) cross product matrix T
- similarly to ANOVA, it can be decomposed into the **Error Sum of Squares and Cross Products E**, and the **Hypothesis Sum of Squares and Cross Products H**

$$= \underbrace{\sum_{\mathbf{i=1}}^{\mathbf{g}} \sum_{\mathbf{j=1}}^{\mathbf{n_i}} (\mathbf{y_{ij}} - \bar{\mathbf{y}}_{\mathbf{i.}})(\mathbf{y_{ij}} - \bar{\mathbf{y}}_{\mathbf{i.}})'}_{\mathbf{E}} + \underbrace{\sum_{\mathbf{i=1}}^{\mathbf{g}} \mathbf{n_i} (\bar{\mathbf{y}}_{\mathbf{i.}} - \bar{\mathbf{y}}_{\mathbf{..}})(\bar{\mathbf{y}}_{\mathbf{i.}} - \bar{\mathbf{y}}_{\mathbf{..}})'}_{\mathbf{H}}$$

Explanation of T, E, H

T

the element **t_k,l** is

$$\sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ijk} - \bar{y}_{..k})(y_{ijl} - \bar{y}_{..l})$$

for k==l

- it is the **total sum of squares** for variable k
- measures the **total variation** in the kth variable,

for k!=l

- measures the **dependence** between variables k and l **across all fof the observations**

E

the element **e_k**,l is

$$\sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ijk} - \bar{y}_{i.k})(y_{ijl} - \bar{y}_{i.l})$$

for k==l

- it is the **error sum of squares** for variable k
- measures the **within treatment variation** for the k-th variable

for k!=l:

- it measures the **dependence between variables k and l** after taking into account the **treatment**

H

the element **h_k,l** is

$$\sum_{i=1}^{g} n_i(\bar{y}_{i.k} - \bar{y}_{..k})(\bar{y}_{i.l} - \bar{y}_{..l})$$

for k==l:

- it is the **treatment sum of squares** for variable k
- measures the **between treatment variation** for the k-th variable

for k!=l:

- measure dependence of variables **k** and **l** across treatments

- consequently, if the **H** is large relative to the **E** we wish to reject H_0

Wilk's lambda test statistics for MANOVA

- the determinant of the error matrix E is divided by the determinant of the total matrix **T = H + E**
- we will reject the null hypothesis if Wilk's lambda is small/close to zero as then **H** is large relative to **E** too

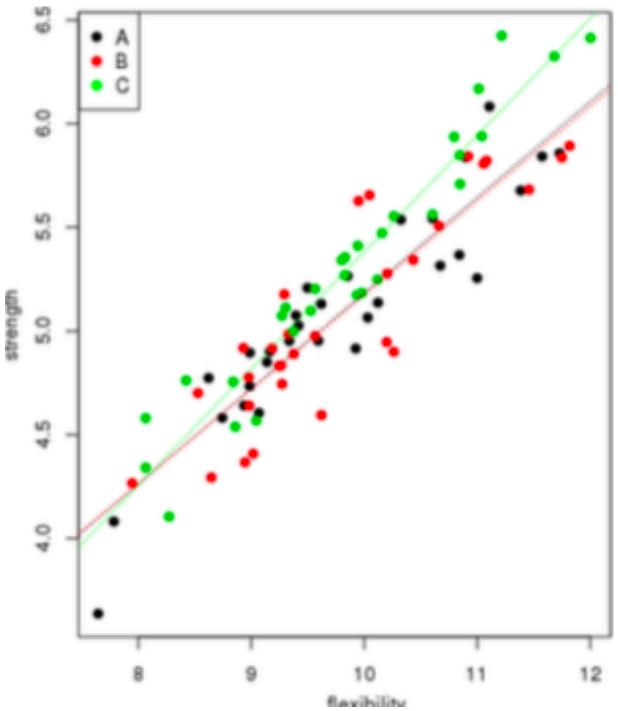
$$\Lambda^* = \frac{|\mathbf{E}|}{|\mathbf{H} + \mathbf{E}|}$$

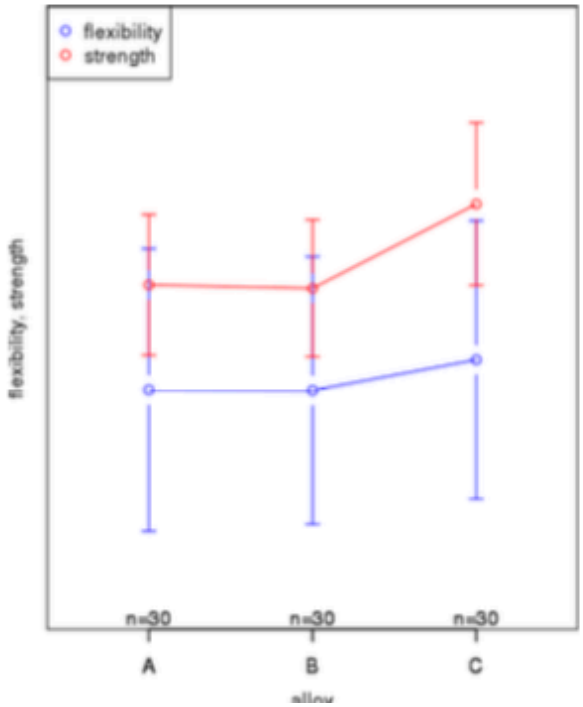
- can also be computed using the eigenvalues ƛ^ of **E^-1 H**

$$\Lambda^* = \prod_{i=1}^{s} \frac{1}{1 + \hat{\lambda}_i}$$

- the distribution of A* (striska) is not tractable, we can only have approximations
- e.g. Barlett;s approximation can be used if N is llarge

$$-\left(N - 1 - \frac{p+g}{2}\right) \ln \Lambda^* > \chi^2_{p(g-1),\alpha}$$





Summary

MANOVA compares multivariate sample means

- it deals with multiple dependent variables at the same time

MANOVA advantages over ANOVA

- better chance to discover which factor is truly important
- protects against Type 1 errors in multiple independent ANOVA runs
- increased power, it can reveal differences not discovered by ANOVA tests

MANOVA cautions

- a complicated design, more difficult to disambiguate
- one degree of freedom is lost for each dependent variable that is added
- unsuitable if the dependent variables are perfectly correlated or uncorrelated

typicaly followed by significance tests on individual dependent variables

04-Regression

Agenda

Linear regression

- simple model with a single predictor
- parameters, interpretation, hypotheses testing
- generalization towards multiple linear ression
- special issues: qualitative predictors, outliers, collinearity

Linear model selection and regularization

- subset selection
- regularization = skrinkage, lasso, ridge regression
- choosing the optimal hermodel, estimating test error

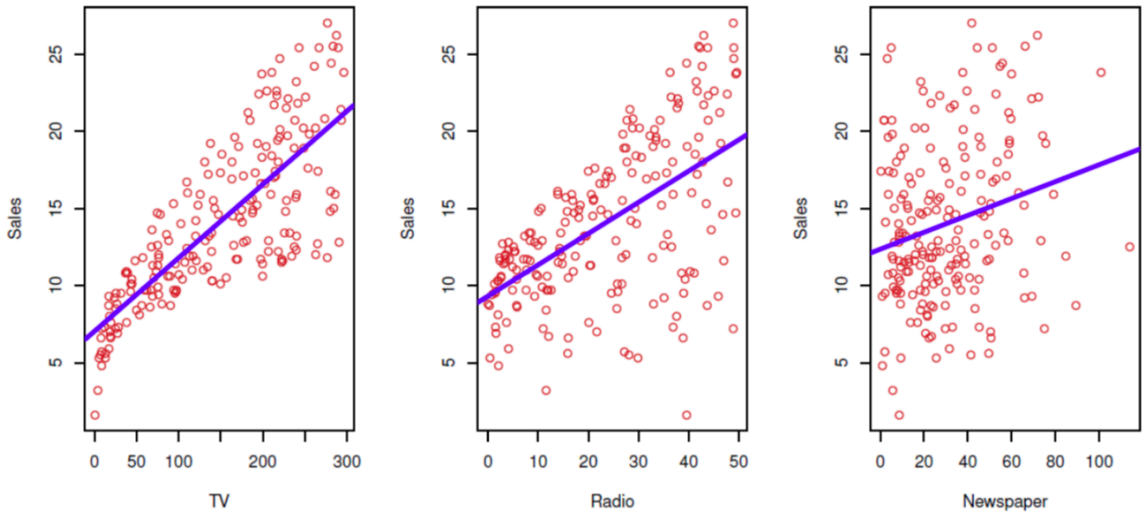
Moving beyond linearity

- polynomial regression
- step funcitons, splines
- local regression
- generalized additive models

Linear regression

Assumption of linearity

- often simpliying assumption only (rarely linear in real world)
- the simplification increases ability to learn and brings interpretability
- good performance preserved in case of moderate violation - otherwise lin models can be extended



Simple linear regression using a single predictor X

We assume a model

$$Y = \beta_0 + \beta_1 X + \epsilon$$

- where β0 and β1 are the unknown constants (coefficients, parameters)
- they represent the **intercept** and **slope**
- ε is the error term

we predict the future values of independent varibale using

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

- where the hat symbol denotes an estimated value
- y^ indicates a prediciton of Y on the basis of X = x

Estimation of parameters by least squares

- ei = y_i - y_i^ represents the ith residual

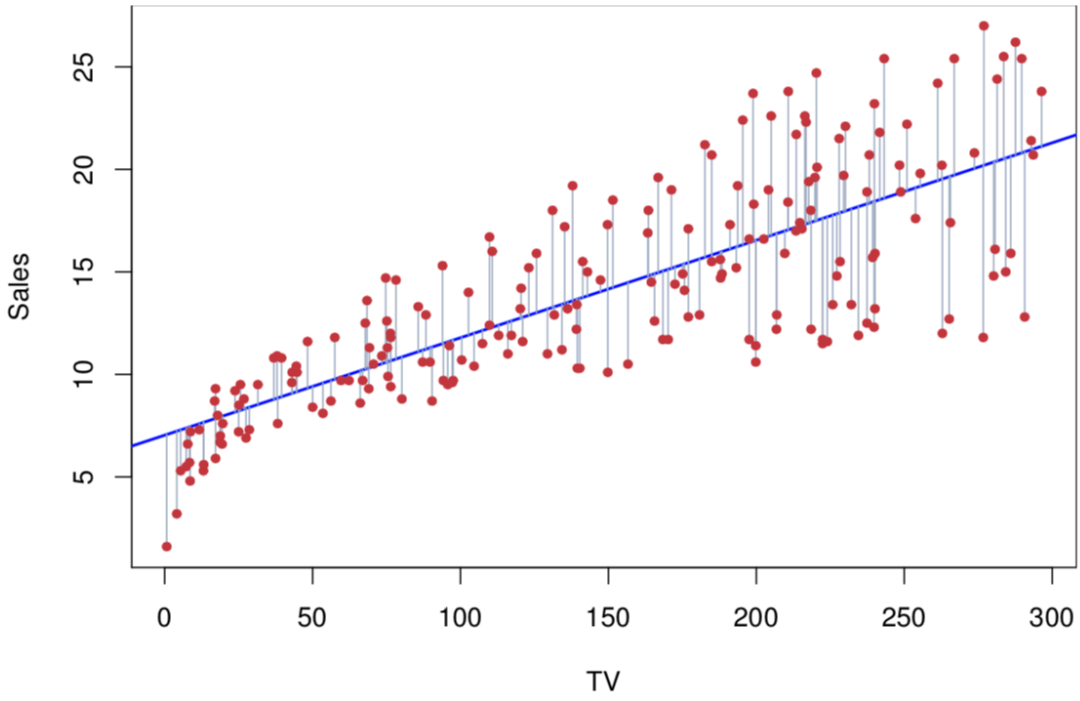
- we define **residual sum of squares (RSS)** as

$$RSS = e_1^2 + e_1^2 + \cdots + e_m^2 = \sum_{i=1}^{m} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$$

- the least squares approach chooses β0^ and β1^ to minimze the RSS

$$\frac{\partial RSS}{\partial \hat{\beta}_1} = 0 \rightarrow \hat{\beta}_1 = \frac{\sum_{i=1}^{m} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{m} (x_i - \bar{x})^2}$$

$$\frac{\partial RSS}{\partial \hat{\beta}_0} = 0 \rightarrow \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$



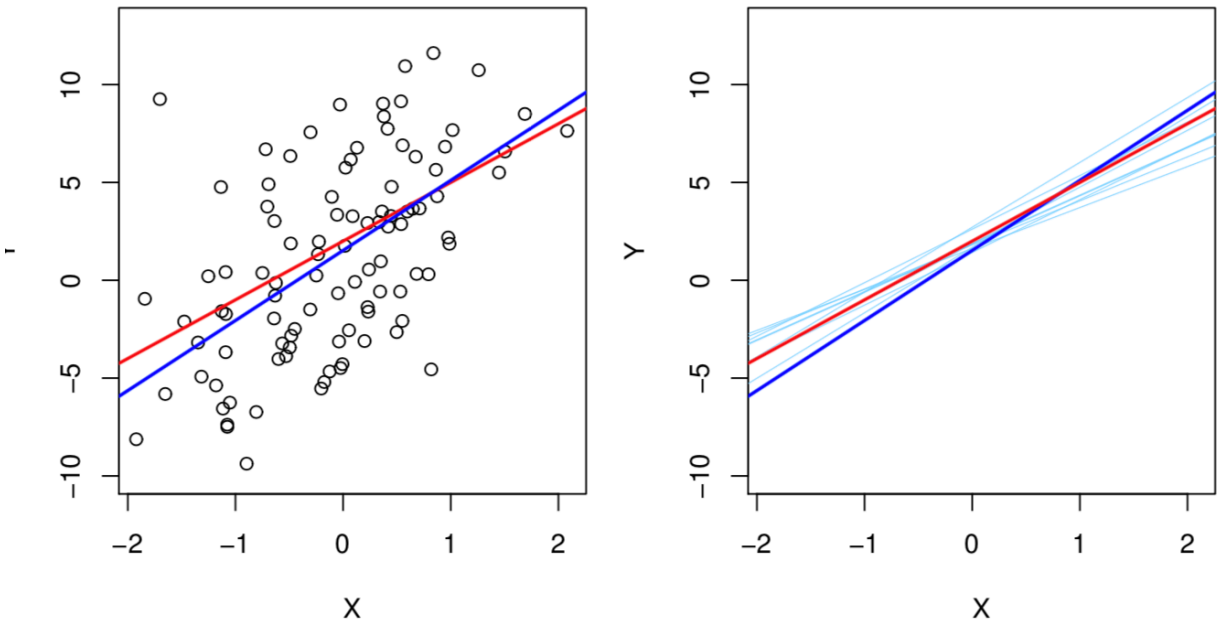
Assessing the accuracy of the coefficient estimates

- **Standard error** of an estimator reflects how it varies under repeated sampling

$$SE(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{m}(x_i - \bar{x})^2} \quad SE(\hat{\beta}_0)^2 = \sigma^2\left(\frac{1}{m} + \frac{\bar{x}^2}{(x_i - \bar{x})^2}\right)$$

- **Residual standard error** RSE is σ(variance) estimate
- gives an absolute measure of its lack of fit

$$RSE = \sqrt{RSS/(m-2)}.$$



Coefficient confidence intervals

Confidence interval (CI)

- 100(1-α)% confidence interval is a range of values that encompasses the true population parameter in 100(1-α)% repeated sampling trials like this

- CI estimate is based on 1-α/2 quantile of a t-distribution with (m-2) degrees of freedom

$$\left[\hat{\beta}_1 - 2SE(\hat{\beta}_1), \hat{\beta}_1 + 2SE(\hat{\beta}_1)\right]$$

$$\left[\hat{\beta}_1 - t_{1-\alpha/2,m-2}SE(\hat{\beta}_1), \hat{\beta}_1 + t_{1-\alpha/2,m-2}SE(\hat{\beta}_1)\right]$$

Hypothesis testing

The most common hypothesis test

- H_0: there is no relationship between X and Y
- H_1: there is some relationship between X and Y

-> mathematically this corresponds to testing

$$- \ H_0 : \beta_1 = 0 \text{ versus } H_A : \beta_1 \neq 0,$$

the test stems from the standard error and t-statistics given by

$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$$

- the corresponding p-value is the probability of observing any value >= than |t|
- both under the H_0 assumption, i.e., assuming β1=0

R-squared

- gives the fraction of variance explained by the model

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

- where **TSS** stands for the **total sum of squares** and **RSS** stands for the **residual sum of squares**

re $TSS = \sum_{i=1}^{m} (y_i - \bar{y})^2$

$RSS = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$ s

- while RSE mentioned earlier gives an absolute measure of its lack of fit

	Coefficient	Std. error	t-statistic	p-value
Intercept	7.0325	0.4578	15.36	< 0.0001
TV	0.0475	0.0027	17.67	< 0.0001

Quantity	Value
RSE	3.26
R^2	0.612
F-statistic	312.1

Multiple Linear regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

- **βj** represents the average effect on Y of a one unit increase in X, **holding all other predictors fixed**
- correlations among predictor cause problems
- the variance of all coefficients tends to increase
- interpretations become hazardous - when Xj changes, everything else changes

Estimation of the parameters

- No principal changes from the simple model
- the prediction formula is

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \cdots + \hat{\beta}_p x_p$$

- the parameter estimates obtained by RSS minimization

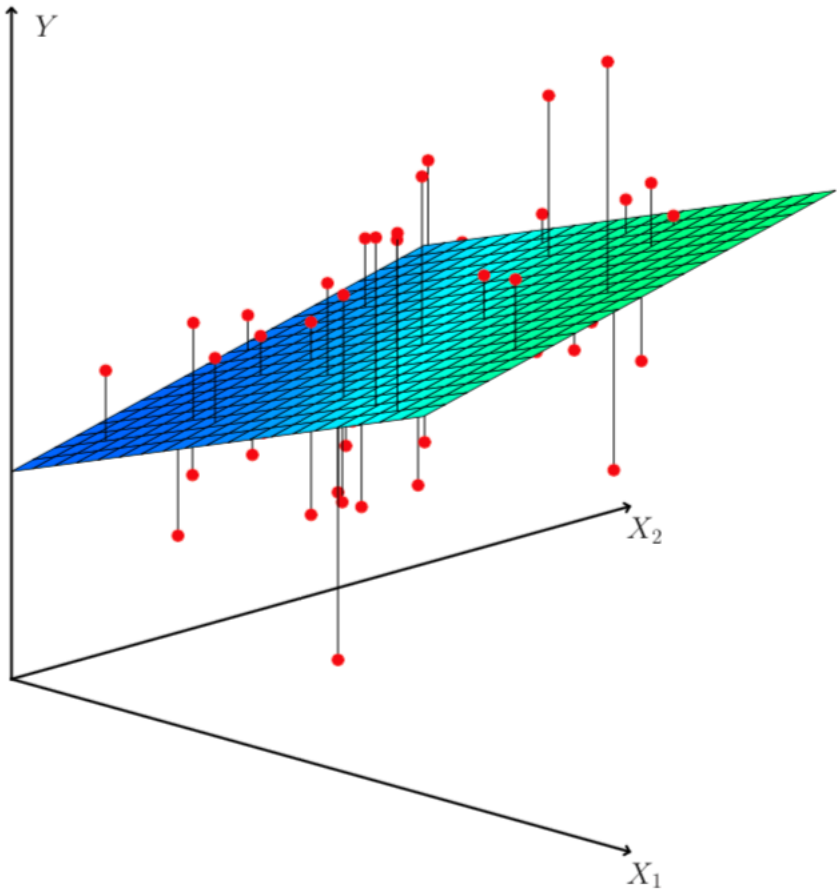
$$RSS = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{m} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \cdots - \hat{\beta}_p x_{ip})^2$$

ordinary least squares estimation

- the most simple approach

generalized least squares

- allow efficient β estimation when heteroscedascity or correlation are present



Is at least one predictor useful?

Formally:

$$H_0: \ \beta_1 = \beta_2 = \cdots = \beta_p = 0 \text{ vs } H_A: \text{ at least one } \beta_j \neq 0,$$

this test is performed by computing the F-statistic

$$F = \frac{(TSS - RSS)/p}{RSS/(m-p-1)}$$

- in fact we compare fit of the full (RSS) and intercept only model (TSS)
- technically, we compute the **ratio between explained and unexplained variance** of the full model
- provided H_0 is true
- E((TSS-RSS)/p) = E(RSS/(m-p-1)) = σ^2 and F is close to 1
- the test is adjusted to the number of predictors p and the sample size m
- F-statistic is compared with quantiles of F(p, m-p-1) distrubution

Deciding on the important variables

- Selection cannot be directly based on the observed predictor p-values
- namely for large p, risk of false discoveries due to multiple comparisons
- build and compare alternative models
- use criterion that balances the training error and model size (e.g. cross-validation)

Direct search methods

all subsets regression

- not feasible -> O(2^p)

forward stepwise selection

- starts with the null model and gradually add the variable that results in the lowest RSS -> O(p^2)

backward stepwise selection

- starts with the full model and gradually removes the variable with the largest p-value in the last model i.e., the least significant one -> O(p^2)
- cannot be used when p>m

Qualitative predictors

Categorical predictors

- takes a discrete set of values

Binary predictors

- we create a new 0/1 variable X_i with the resulting model

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i = \begin{cases} x_i = 1 : \beta_0 + \beta_1 + \epsilon_i \\ x_i = 0 : \beta_0 + \epsilon_i \end{cases}$$

- interpretation:
- β_0 is the average outcome in the zero group,
- β_0 + β_1 is the average outcome in the positive group
- Thus the β_1 is the average difference in outcomes between groups

L-levels predictors

- we typically create l-1 dummy variables

$$x_{i1} = \begin{cases} 1 & \text{if } i\text{th person is Asian} \\ 0 & \text{if } i\text{th person is not Asian} \end{cases} \qquad x_{i2} = \begin{cases} 1 & \text{if } i\text{th person is Caucasian} \\ 0 & \text{if } i\text{th person is not Caucasian} \end{cases}$$

Interaction between variables

- adding an interaction term in term when the problem of the interaction between variables is shown

$$\text{sales} = \beta_0 + \beta_1 \times \text{TV} + \beta_2 \times \text{radio} + \beta_3 \times (\text{TV} \times \text{radio}) + \epsilon$$

	Coefficient	Std. error	t-statistic	p-value
Intercept	6,7502	0.248	27.23	< 0.0001
TV	0.0191	0.002	12.70	< 0.0001
radio	0.0289	0.009	3.24	0.0014
TV×radio	0.0011	0.000	20.73	< 0.0001

- the p-value for the interaction term TVxradio is extremely low - indicating that there is strong evidence for H_a: B_3 != 0
- R^2 for the interaction model is 96.8%
- compared to only 89.7% for the model using TV and radio without an interaction term
- 69% of the variability after fitting the additive model has been explained by the interaction term
- if we include an interaction term, we should also include the main effects (TV and radion) even if their p values are not significant

Choosing an optimal model

- three classes of methods

Subset selection
- repetetive application of least squares on various reduced sets of predictors

Dimension reduction
- ordinary least squares regression in a L-dimensional subspace

Shrinkage (regularization)
- we fit a model involving all p predictors, but the estimated coefficients are shrunken towards zero
- effect of reducing variance, can also perform variable selection

RSS and **R^2** are **not suitable** for selecting the best model
- they are related to the training error
- the model containing all of the predictors will always have the smallest RSS and the largers R^2

Adjusted R-squared
- unlike the R^2 statistic, it pays a price for unnecessary predictors
- for a least squares model with p variables

$$\text{Adjusted } R^2 = 1 - \frac{RSS/(m-p-1)}{TSS/(m-1)}$$

- a maximization criterion
- a heuristic criterion
- irrelevant variables bring a small decrease in RSS, this decrease is outweighted by decrease in m-p-1 (neither TSS nor m changes with p)

Shrinkage methods

Ridge regression

- should be applied after standardizing the predictors

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}}$$

- recall that the least squares procedure minimizes RSS

$$RSS = \sum_{i=1}^{m}(y_i - \hat{\beta}_0 - \sum_{j=1}^{p}\hat{\beta}_j x_{ij})^2$$

- in contrast, the ridge regression coefficient estimates are the values that minimizes

$$\sum_{i=1}^{m}(y_i - \hat{\beta}_0 - \sum_{j=1}^{p}\hat{\beta}_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\hat{\beta}_j^2 = RSS + \lambda\sum_{j=1}^{p}\hat{\beta}_j^2$$

- where lambda >= 0 is a **tuning parameter**, to be determined separately, typically CV is used

$$\lambda \sum_j \hat{\beta}_j^2$$

is a **shrinkage penalty** with the effect of shrinking the β_j estimates towards zero

Bias variance trade-off

- suppose we have fit a model f^(x) to some training data
- let (x0,y0) be a test observation drawn from the same population
- if the true model is Y = f(X) + eps then

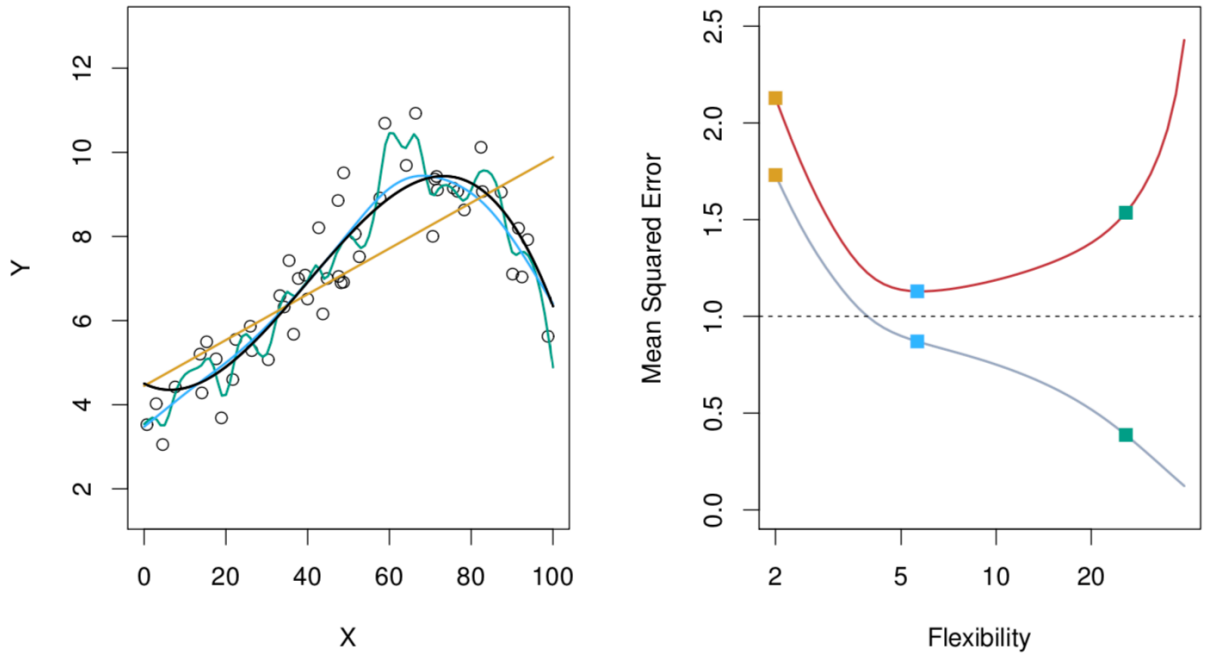
$$E\big(y_0 - \hat{f}(x_0)\big)^2 = Var(\hat{f}(x_0)) + \big[Bias(\hat{f}(x_0))\big]^2 + Var(\epsilon)$$

$$Bias(\hat{f}(x_0)) = E(\hat{f}(x_0)) - f(x_0)$$

- the error can be decomposed into model **variance, bias** and irreducible error
- as the flexibility of f^ increases, its variance increases, its bias decreases
- choosing the flexibility based on average test error amounts to a bias variance trade-off

Relationship between model flexibility and accuracy

- and the bias of the train error towards more flexible/overfit models



red curve is generalization err, grey curve is training err

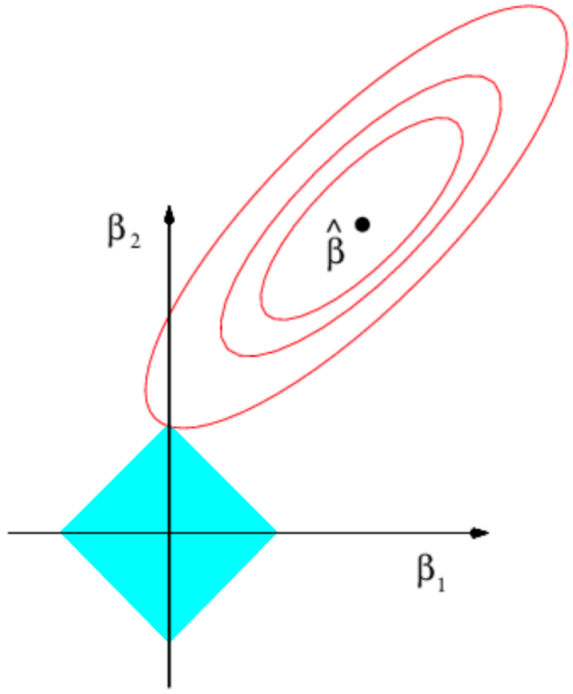
The lasso

- ridge regression will include all p predictors in the final model
- disadvantage, does not help in feature selection
- lasso overcomes this problem by minimizing

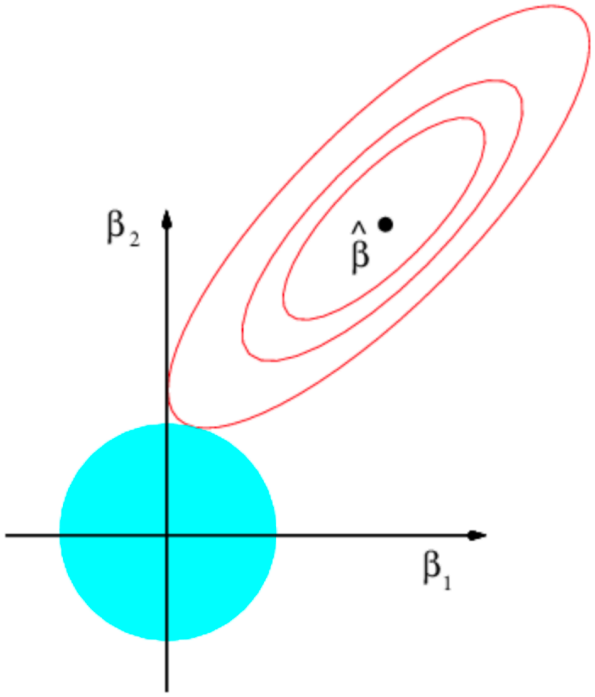
$$\sum_{i=1}^{m}(y_i - \hat{\beta}_0 - \sum_{j=1}^{p}\hat{\beta}_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\hat{\beta}_j| = RSS + \lambda\sum_{j=1}^{p}|\hat{\beta}_j|$$

- uses **l1 penalty** instead of l2 penalty a
- the l1 penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when λ is sufficiently large
- lasso yields **d**and performs **variable selection**

Lasso vs ridge regression



Lasso



Ridge regression

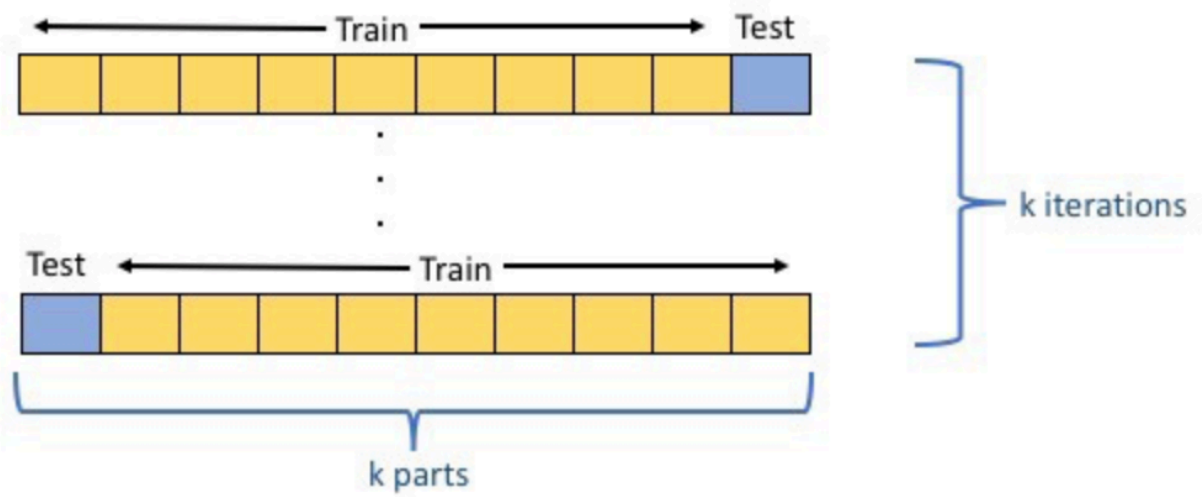
Selection of tunning parameter value

- which λ value is the best
- employ **cross validation**

Cross-validation

- the training erro can dramatically underestimate the test error
- **hold-out** makes the most easy approach to model testing
- split the data between train and validation set 70:30
- sufficient for large data sets
- k-fold cross-validation

- training on k-1 folds and testing on the remaining fold (always different)



Summary

Multiple linear regression
- a simple model with the strong assumption of linearity
- helps to understand concurrent effects on a target continous variable

Model selection and regularization may improve prediction and understanding
- neither ridge regression nor the lasso will universally dominate the other
- **lasso** performs better when the response is a function of only a relatively **small number of predictors**
- however the number of predictors related to response is never known a priori
- cross-validation can help to decide which approach is better on a particular data set and select a value of the tuning parameter

05-Discriminant Analysis

Introduction

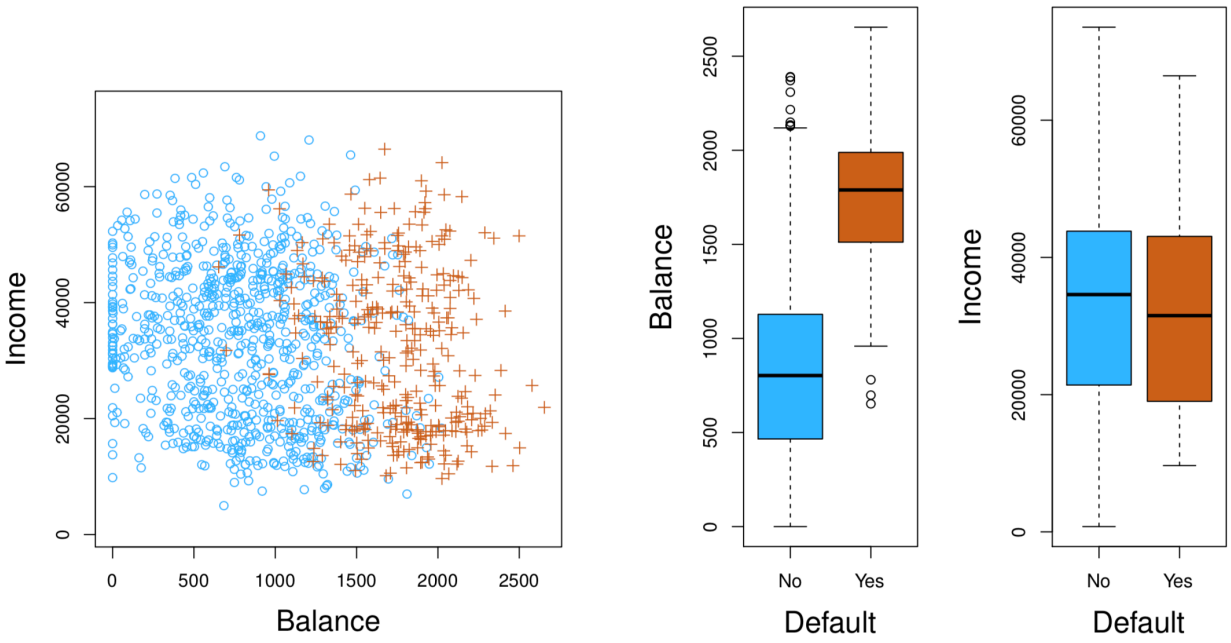
Study multivariate relationships with **categorical dependent variable**

- **independent variables** are **continuous** (can be categorical too)
- **nominal** dependent variables take values in an unordered set C

$$\text{color} \in \{\text{brown, blue,green}\}, \ \text{email} \in \{\text{spam, ham}\}$$

Main goals

- **classify** into the target categories
- **understand** the role of the individual independent variable



Can we use linear regression?

$$Y = \begin{cases} 0 & \text{if No} \\ 1 & \text{if Yes} \end{cases}$$

perform a linear regression of Y on X and classify as Yes if Y^ > 0.5

- in the case of a binary outcome linear regression works well as a classifier, is equivalent to **LDA**

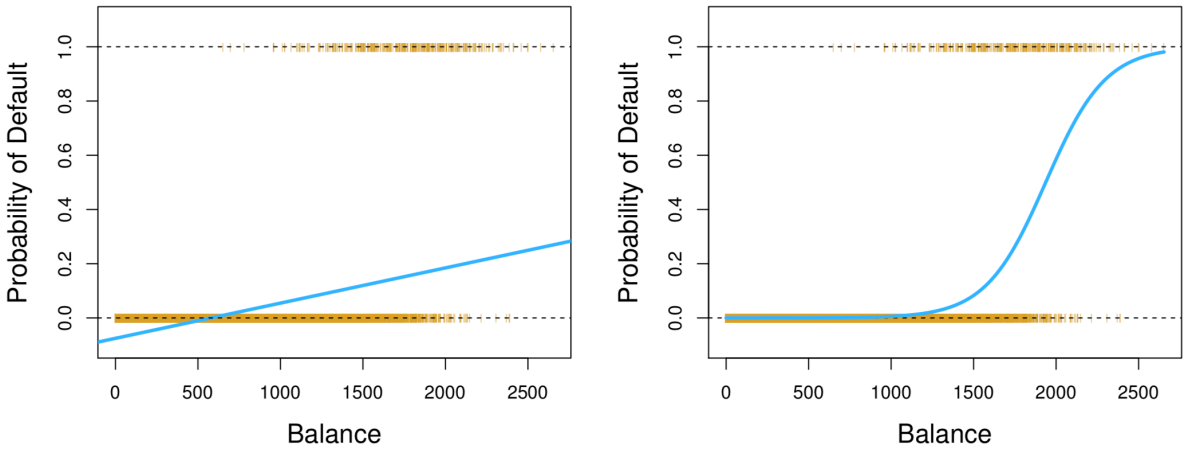
- however linear regression migh in general
- produce probabilities less than zero or bigger than one
- be sensitive to outliers
- “mask out” some classes in problems with multinomial targets

Logistic regression is more appropriate

Logistic regression

- introduce a non-linear **logit** transformation
- p(y=1|**x**) = p(**x**) = exp(x)/ 1+exp(x)
- this monotone transformation is called the **log odds** or **logit** transformation of p(X)

$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\ldots\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\ldots\beta_p X_p}} = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\ldots\beta_p X_p)}}$$



We use **Maximum Likelihood** to estimate the parameters β_i

$$\ell(\beta_0,\ldots,\beta_p) = \prod_{\forall i\, y_i=1} p(\mathbf{x_i}) \prod_{\forall i\, y_i=0} (1-p(\mathbf{x_i}))$$

Using as predictor

- compare log-odds for two different groups

$$\frac{p(s^+)}{1-p(s^+)} = e^{\hat{\beta}_0+\hat{\beta}_1} \;\&\; \frac{p(s^-)}{1-p(s^-)} = e^{\hat{\beta}_0} \rightarrow \frac{\frac{p(s^+)}{1-p(s^+)}}{\frac{p(s^-)}{1-p(s^-)}} = e^{\hat{\beta}_1}$$

e^β1 gives the **odds ratio** between the groups

Using for multiple (k>2) classes (Multinomial regression)

- easily generalizable for multiple classes
- for each class there is a linear function

$$Pr(Y = k|\mathbf{X}) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+\ldots\beta_{pk}X_p}}{\sum_{j=1}^{K} e^{\beta_{0j}+\beta_{1j}X_1+\cdots+\beta_{pj}X_p}}$$

K-1 models are trained

$$\forall i = 1 \ldots K-1 \quad \frac{Pr(Y = i|\mathbf{X})}{Pr(Y = K|\mathbf{X})} = e^{\beta_{0i}+\beta_{1i}X_1+\cdots+\beta_{pi}X_p}$$

it can easily be shown that

$$Pr(Y = i|\mathbf{X}) = \frac{e^{\beta_\mathbf{i}\cdot\mathbf{X}}}{1+\sum_{j=1}^{K-1} e^{\beta_\mathbf{j}\cdot\mathbf{X}}} \qquad Pr(Y = K|\mathbf{X}) = \frac{1}{1+\sum_{j=1}^{K-1} e^{\beta_\mathbf{j}\cdot\mathbf{X}}}$$

Discriminant analysis

- the distribution of X in each of the classes modeled separately
- Bayes theorem helps to obtain P(Y|X)

$$Pr(Y = k|\mathbf{X} = \mathbf{x}) = \frac{Pr(\mathbf{X} = \mathbf{x}|Y = k)Pr(Y = k)}{Pr(\mathbf{X} = \mathbf{x})}$$

- when we use normal (gaussian) distributions for each class
- this options leads to **linear or quadratic discrimination analysis**

$$Pr(Y = k|\mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{j=1}^{K} \pi_j f_j(\mathbf{x})}$$

- where $f_k(\mathbf{x}) = Pr(\mathbf{X} = \mathbf{x}|Y = k)$ is the density for $\mathbf{X}$ in class k,
- where $\pi_k = Pr(Y = k)$ is is the marginal or prior probability for class k.

LDA for p=1

- plug gauss dist into bayes formula

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{j=1}^{K} \pi_j \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_j}{\sigma}\right)^2}}$$

there are simplifications

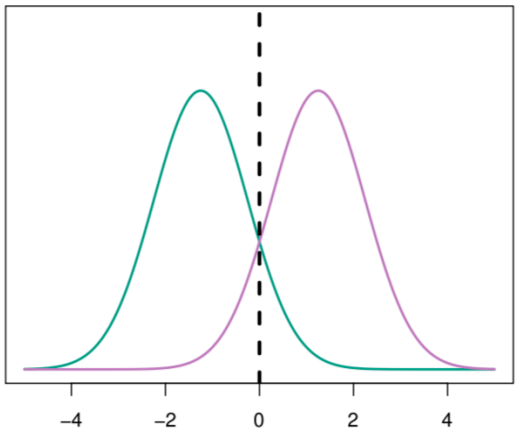
- maximize the **discriminant score** instead

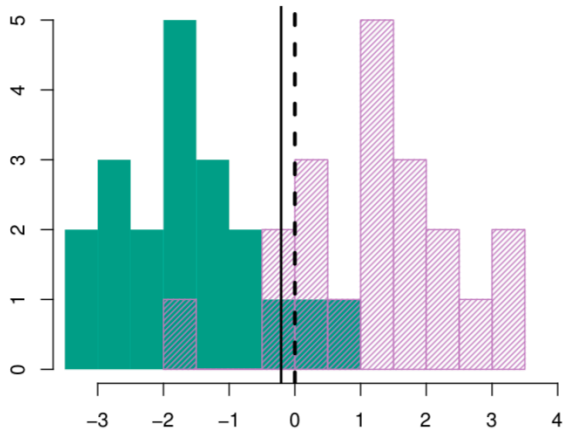
$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + log(\pi_k)$$

- linear function of **x**

Estimating the parameters

- typically these parameters are unknown, we estimate them from data
- example μ1 = -1.5, μ2 = 1.5, π1 = π2 = 0.5 and σ^2 = 1





LDA for p>1

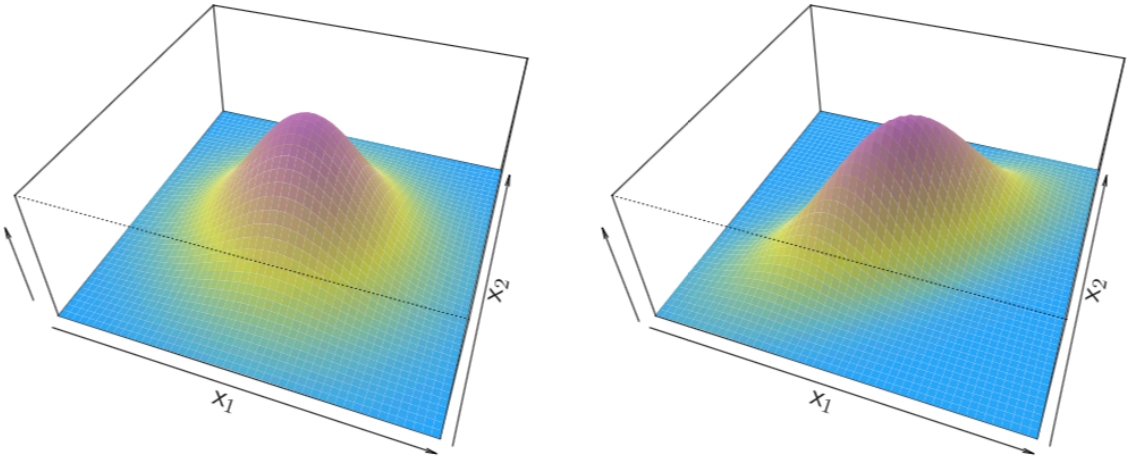
Density

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)}$$

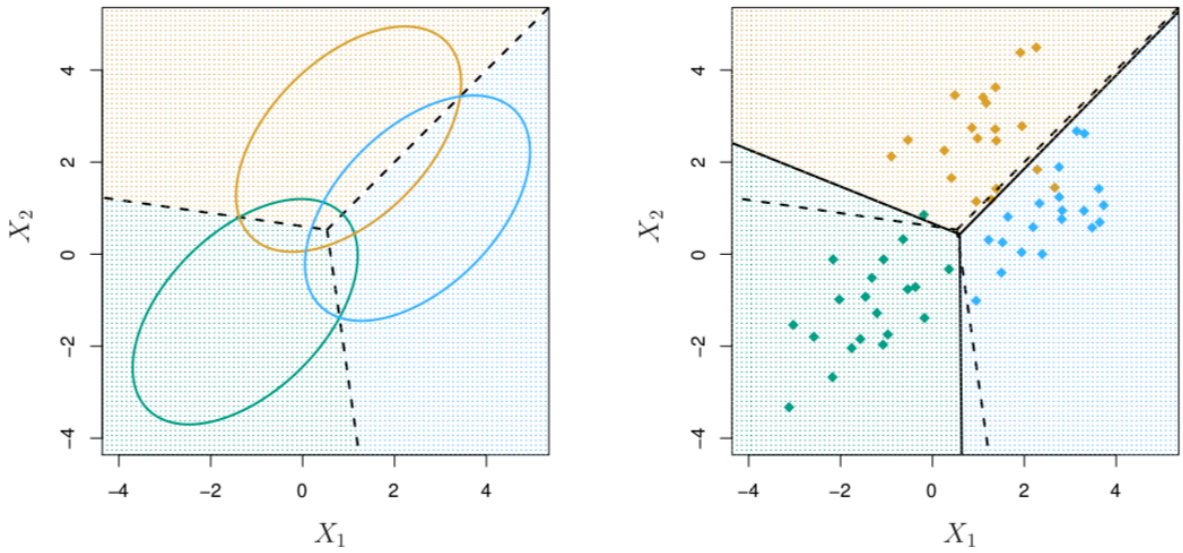
Discriminant function

$$\delta_k(\mathbf{x}) = \mathbf{x}^T\mathbf{\Sigma}^{-1}\mu_k^T - \frac{1}{2}\mu_k\mathbf{\Sigma}^{-1}\mu_k + log(\pi_k),$$

- despite complexity, it's a linear function of x too



example: p=2, k = 3



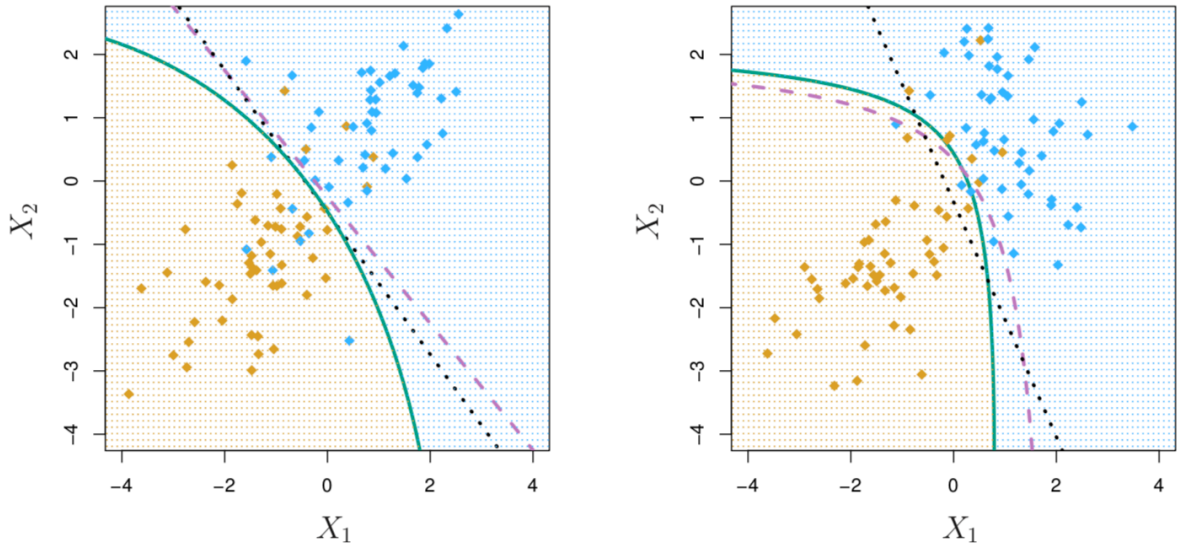
From δ_k(x) to class probabilities

- turn discriminant score into class probability estimates

$$\hat{Pr}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{K} e^{\hat{\delta}_l(x)}}$$

- classifying to the largest δ_k(x) amounts = for which P(Y=k|X=x) is largest

Quadratic Discriminant Analysis



- puprle dashed line - Bayes optimal boundary, LDA black, QDA green

- left: cov matrices truly match
- LDA is close to optimal solution
- QDA sufffers from higher variance
- right: the orange class has a positive correlation between predictors, blue class negative, class cov matrices differs
- optimal boundary is quadratic
- LDA suffers from higher bias

Summary

Logistic regression is very popular for classification, especially when **K=2**

LDA is useful when:
- the number of samples is small or the classes are well seperated
- gaussian assumptions are reasonable

Naive Bayes is useful when:
- the dimension is very large

K-NN is useful when:
- the parametric assumptions used above do not hold
- the decision boundary is highly non-linear
- disadvantage: no direct outcome on feature importance

No methods dominates each others in every situation