Applied Statistics

A complete transcript of lectures.

Always consider checking information from this document, as it may contains errors.

Course held by Piercesare Secchi

Politecnico di Milano

AY. 2023/2024

Disclaimer: it is not an official document of the course, should not be seen as replacement of lectures and exercise sessions, but as supporting material.

Dalla gente per la gente

Giuseppe Gentile

22 Feb

Statistical learning: use training set to build a function to explain the variability of y in terms of X.
y is a random variable, and $\underline{x}$ is a random vector:

$$f: R^p \to R \quad s.t. \quad f(\underline{x})\ predict\ y$$

The space of random variables is *R* and contains a subspace with all random variables that you can get once you have x (the output of $f(\underline{x})$).

We have another random variable: **y**.

- If it lives in the subspace generated by $f(\underline{x})$, then you know everything about y: but is not interesting, is like having x as the temperature in Celsius and y the temperature in Kelvin, there is a deterministic law.
- The usual case is that y doesn't lives in the subspace generated by the function, but in the space of random variables.

Goal: find the function that minimize the distance between y and $f(\underline{x})$.

$$f: R^p \to R \quad s.t. \quad min\ |y - f(\underline{x})|^2$$

y and $f(\underline{x})$ are both numbers. Taking the square because enlarge bigger errors and give advantage to smaller ones.
We want to minimize this error with **all the possible x-s,** not only for training set, but for all the realizations of y and x.
Therefore, we have to change a bit the problem, taking the mean, considering all the possible realizations:

$$f: R^p \to R \quad s.t. \quad min\ E[|y - f(\underline{x})|^2]$$

To minimize this distance between those two vectors, project the y on the space of the function (call this space $\sigma(x)$).
The projection is $\pi_{y|\sigma(x)}$, and must be the closest as possible to $f(\underline{x})$.

The best projection is called **regression function**:

$$\pi_{y|\sigma(x)} = E[y\ |\underline{x}]$$

Note: is not the linear regression function, that is a specific case of regression function.

The general model for statistical learning is:

$$y = f(\underline{x}) + \varepsilon, \text{ where } f(\underline{x}) = E[y\ |\underline{x}]$$

$\boldsymbol{\varepsilon}$ is a random variable **orthogonal to** $\boldsymbol{\sigma(x)}$ with $E[\varepsilon] = 0$.
Depends on the model you chose.
Note: not zero, but its average is 0. And represent what miss to explain y in terms of x.

In statistical learning you have to learn f from data (and from any knowledge you have on the problem).

K Nearest Neighbours

Let $N_x = \{k\ x_i\ in\ training\ set\ close\ to\ x\}$

$$f(x) = \frac{1}{k} \sum_{x_i\ in\ N_x} y_i$$

Is a local method -> curse of dimensionality.
Being close has sense only if you live in small space. Once the space gets big, you are alone.

We must reduce the dimensionality of the problem:

1- Reduce the number of features p (PCA)
2- Reduce the dimensionality of function: look only to some subset of functions, and not every possible functions -> **structured models:** find f having some parameters to be chosen.
   Call θ the parameters: $\theta = (\beta_0, \dots, \beta_p)$

$$f_\theta(x) = \beta_0 x_0 + .. + \beta_p x_p$$

   This is a **linear model.** However structured models in general can have any form.

Generalization error
Call $\hat{f}(x)$ the estimate:

$$E_x = E\left[\left(y_0 - \hat{f}(x_0)\right)^2\right] = \underbrace{\left(f(x_0) - \hat{f}(x_0)\right)^2}_{\text{Reducible error}} + Var(\varepsilon_0)$$

Error:

$$E[E_x] = \underbrace{\left(f(x_0) - E[\hat{f}(x_0)]\right)^2}_{Bias} + \underbrace{Var\left(\hat{f}(x_0)\right)}_{Variance} + \underbrace{Var(\varepsilon_0)}_{Irreducible}$$

The more the model is complex, the lower the bias, but the more the variance.
The less complexity in the model, the lower the variance, but high biased.
Overfitting when low bias but extremely high variance.
Being unbiased is **good if you have low variability**: if you park the car 50% of the times into a tree and 50% into the other tree, on average, you're parking exactly in the middle. The zero bias is telling us that we're doing good, but we're not parking anytime in the middle!

26 Feb

Covariance: indicates the direction of the linear relationship between two variables. If is positive, it means that when a variable increase the other increase. If negative if one increase the other decrease. However, is difficult to interpret the scale of the increasing/decreasing.

Correlation: face the issue of the covariance by standardizing it. In this way you have values between [-1,1].

27 Feb

$x_1, \dots, x_n$ random vectors in R^p. If they are iid, every unit has the same probability to be chosen.
Let fix a generic distribution, for which the mean and covariance are the unknown, and we want to find them from data (looking for estimators).
However, this does not allow to know the distribution in the general case (it is enough only if the distribution is gaussian).

- The estimator of the mean is the **average**. This to be true **only needs** to have **identical distribution**, and not necessarily independent.
    - Is **unbiased**: on average what you estimate is right.

- The **estimator of the covariance** is $S_n = \frac{1}{n}\sum_i (x_i - \bar{x})(x_i - \bar{x})^T$. This instead **needs both independence**, for different units **and identical distribution** for same sample (i=j).

    - Is **biased**, it underestimates the variability: $E[S_n] = \frac{n-1}{n}\sum$.

    - To make it unbiased take $S = \frac{n}{n-1}S_n$. In this way, $E[S_n] = \sum$.

Multivariate variability

I have my linear space of the $\underline{1}$ vector. My column vector of the data frame is called c_1(say the salary of different persons).
The linear space of the $\underline{1}$ vector is where there is no variability.
The closest vector to the linear space of $\underline{1}$ vector is the orthogonal projection of c_1 on the linear space generated by $\underline{1}$.
c_1 can be find by Pitagora, sum of square of orthogonal projection and $\underline{1}$ vector.

Call d_1 the vector that project c_1 to the linear space generated by vector $\underline{1}$.

$$d_1 = c_1 - \pi_{c_1|\,1}$$

Apply this for every column, and you get the deviation matrix. $d \in R^{n\,x\,p}$.

$$d = \left[I - \frac{11^T}{1^T 1}\right] X$$

From this you get the **S matrix**: **sample variance**. This makes explicit the column perspective.

$$S = \frac{d^T d}{n-1} = by\ idempotency\ and\ simmetricity = \frac{1}{n-1} X^T \left[I - \frac{11^T}{1^T 1}\right] X$$

If the angle between two column vector is zero, then once I know a feature, I can get the other too: they are correlated.
If is $\pi/2$, they are independent.

We need to capture multivariate variability with a single number.
There are two ways:

- **Generalized variance**: Det(S). Consider the relationships of variables. Is useful to capture the variance in a multivariate dataset. Is the **area** of the parallelogram generated by the two

deviation vectors of the two variables. Consider both the angle and the norm of them.
To increase this area, you must increase the norms of the two vectors or the angle.
Note: has maximum value when the two vectors are not correlated, so orthogonal, $\theta = \frac{\pi}{2}$.
Is minimum when they are linear dependent.
So this **tells** us **both about** their **variability** (the norm of the vectors) **and their correlation**.

- $Det(S) = 0$ is 0 **IFF** deviation vectors are linearly dependent. So you can express at least one feature as linear combination of the others. Is only redundancy in the dataset, is not good, even if you are perfectly fitting the data. Like measure two things at the same moment with different unit, useless. This is the case also in the case of determinant close to 0.
- **IF** $p \geq n \implies Det(S) = 0$. So if you observe more feature than units, the generalized variance is zero. You'll find a perfect linear relationship but is not significant to our analysis. Take two people's height and weight and you'll find a perfect line that cross both of them. The reason of this is the **bias-variance trade-off.**

$$Det(S) = \prod_{i}^{p} \lambda_i$$

Total variance: Trace(S). Doesn't take into account the dependence between different variables.
Tells us about the perimeter: it doesn't change if you change the angle, so doesn't consider their correlation.

$$Trace(S) = \sum_{i}^{p} \lambda_i$$

If $S \in R^{pxp}$ is symmetric, for the spectral decompositions exists p eigenpairs. Eigenvectors are orthogonal in the space of p dimension.
Let **P** be the matrix with **eigenvectors in column**, and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \dots, \lambda_n)$.
Ordered such that $\lambda_1 \geq \cdots, \geq \lambda_n \geq 0$
In general, S is semipositive definite ($\lambda_i \geq 0 \; \forall i$)
But if $Det(S) \neq 0 \implies S \; is \; positive \; definite$.

- If this holds, exists S^{-1}. Every eigenvalue is strictly positive. We can define a new distance metric: **Mahanalobis Distance**. $d_{S^{-1}}(x, y) = \sqrt{(x-y)^T S^{-1}(x-y)}$. I'm modifying the geometry of the distance to take into account the variability. If $S = I \implies$ is the Euclidean distance.
  A distance must be **positive, symmetric and satisfy triangular inequality.**
  In the Euclidean space, $E_r = \{x \in R^p : d^2_{S^{-1}}(x, \bar{x}) \leq r^2\}$ is an ellipse zero centred with radius inversely proportional to the square root eigenvalues of S **(note, not of S^{-1})**.
  The volume of the ellipse is $\frac{r^p}{\sqrt{\lambda_1}\sqrt{\lambda_2}\cdots\sqrt{\lambda_p}}$.

Mahanalobis capture the distance in the space induced by the variability, not in terms of unit of measure as Euclidean distance.

The distance of a point from the mean of the distribution grows along each principal component.
If there isn't variance, you go back to Euclidean distance case, where is the same to move in every direction you chose.

29 Feb

$$d_{S^{-1}}: R^p x\ R^p \rightarrow [0, \infty)$$
$$d^2_{S^{-1}}\left(\underline{x}, \underline{y}\right) = \left(\underline{x} - \underline{y}\right)^T S^{-1}\left(\underline{x} - \underline{y}\right)$$

If we have $S = \begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix}$, with $S_{11} > S_{22}$. This means that first distance is smaller than second.

The ellipse:

$$E_1 = \{\left(\underline{x} - \underline{y}\right)^T S^{-1}\left(\underline{x} - \underline{y}\right) \leq 1\}$$

The volume of this ellipse is proportional to $\sqrt{\det(S)} = \sqrt{\lambda_1 - \lambda_2}$

If S_{11} increases or S_{22} increases the volume increases too.
Mahanalobis distance consider the variability of data when calculating the distance between different units.

In general case in R^p, the covariance is different than 0. The ellipse in this case is rotated and not aligned with the axis.

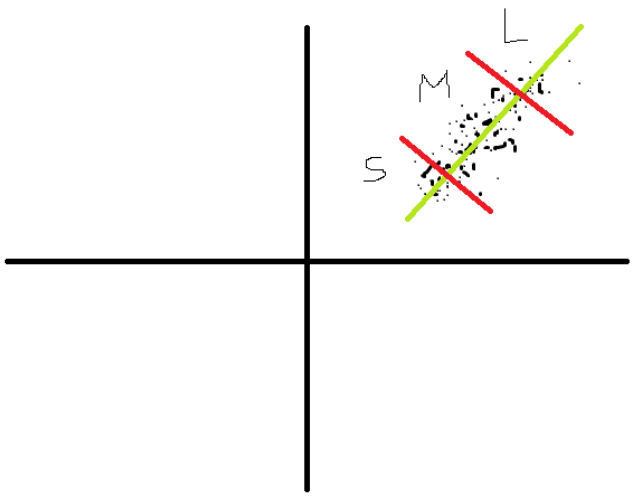
- ➔ Choose a different reference system, the one spanned by the eigenvectors of S:
  $S = \sum_{i=1}^{p} \lambda_i e_i e_i^T = P\Lambda P^T$
  (λ_i-s ordered decreasing).
  In this way we find a reference system with diagonal covariance: spectral decomposition.

With quantitative variables:
Correlation isn't related to data, but with reference system (example of stars)

- ➔ I can find a reference system where variables are uncorrelated

In this new reference system, generalized variance and total variance doesn't change.

Plotting with respect to height and weight of a person:



PCA finds the direction that maximize the variability, in a setting with lots of variables.

Find $\underline{a} \in R^p, ||\underline{a}|| = 1$, st. $Var(\underline{a}^T \underline{X})$ is max, and $Cor(\underline{a}_i{}^T \underline{X}, \underline{a}\ \underline{X}) = 0$ for $i = 1, \dots, p-1$.
This for each feature.

$\underline{e}_1$ **loadings of first principal component**, direction where we have maximal variability. First eigenvector of Σ, which is the $Cov(\underline{X})$.

$y_1 = \underline{e}_1 \underline{X}$ **score** of the first principal component. Combinations with some weights of the original features

This for each feature: $y_i = \underline{e}_i \underline{X}$

Uncorrelation is equivalent to orthogonality: eigenvectors. Principal components are uncorrelated.

$$Cov(y_i, y_j) = \begin{cases} 0 & if\ i \neq j \\ Var(y_i) = e_i^T \Sigma e_i = e_i^T \lambda_i e_i = \lambda_i & if\ i = j \end{cases}$$

There is an order in principal component: the first is the one with largest variability, the second is with the second largest variability and so on.

Typically, you centre the new dataset on the mean, since doesn't impact covariance $y_i = \underline{e}_i(\underline{X} - \underline{\mu})$.
With PCA we rank individuals along the maximum variability.
Generalized variance and total variance don't change.

4 Mar

Principal component means represent X into a different reference system.
This new reference system is given by the eigenvectors of the covariance matrix **Σ**.
Remark: $\Sigma = \mathrm{P}\Lambda\mathrm{P}^{\mathrm{T}}$

Let y be the new vector in the new (centred) reference system.

$$\underline{y} = P^T(\underline{X} - \underline{\mu})$$

Therefore:

$$E\left[\underline{y}\right] = 0$$

$$Cov\left(\underline{y}\right) = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_{\mathrm{p}} \end{bmatrix} = \mathbf{\Lambda}$$

Remark: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{\mathrm{p}}$, with $\lambda_{\mathrm{i}} = Var(y_i)$

Therefore, the proportion of variability explained by the $j-th$ component is:

$$\frac{\lambda_{\mathrm{j}}}{\sum \lambda_{\mathrm{i}}}$$

The total variability is $\sum \lambda_{\mathrm{i}}$ that is: $tr(\Lambda) = tr(\Sigma)$.
Typically, what is done is, choose *k* such that the total variability is greater than the threshold.

$$\frac{\sum_j^k \lambda_j}{\sum \lambda_i} \geq threshold$$

The first component has the highest jump in the variability explained.

Or, you can choose the amount of components to take based on the **elbow**, so, take components until you have the plot that is going more vertically.

Main criticality of PCA: interpretation of components is hard. The new component is a linear combination of the original variables (centred here)

$$y_i = \underline{e_i}^T \left(X_i - \underline{\mu}\right) = e_{1i}(X_1 - \mu_1) + e_{2i}(X_2 - \mu_2) + \cdots + e_{pi}(X_p - \mu_p)$$

But what is the correlation between original variables (x) and the new ones (y)? **How can I explain the new variables in terms of the original ones?**

$$Corr(y_i, x_k) = e_{ki}\frac{\sqrt{\lambda_i}}{\sqrt{\sigma_{kk}}}$$

- ➔ if all the variances would be similar (all components of x have similar variances $\sqrt{\sigma_{kk}} = 1$), the correlation is proportional to the loadings $Corr(y_i, x_k) \propto e_{ki}$.
  e_{ki} is the k-th row of the i-th column of the P matrix.

The higher the loading the higher the correlation with that variable, so considering the variable with high correlation may be interesting.
But this interpretation can be done only if the variances of the components is similar, if they are very different for different variables, this interpretation fails.

Write again in a better form: **Interpretation of the PCA**
If $\sigma_{11,}\sigma_{22}, \dots, \sigma_{kk}$ are of similar order of magnitude → I can explain the new variable in terms of the original ones basing on the loadings.

So, **if $\boldsymbol{\sigma_{11,}\sigma_{22}, \dots, \sigma_{kk}}$ are very different**, then **standardize dataset** first.

This because if the variances are very different, PCA will produce just another reference system that is the same of the original one with new axis, because will see the variable with the highest variable, masking all the other variables.

Standardizing allows to get rid of the unit of measure, allowing to see how far are you are from the standard deviation.
Standardizing:

$$Z = V^{-1/2}\left(\underline{X} - \underline{\mu}\right) \qquad where\ V = \begin{pmatrix} \sigma_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{kk} \end{pmatrix}$$

$$E[Z] = 0 \qquad Cov(Z) = V^{-1/2}\,\Sigma\,V^{-1/2} = \boldsymbol{\rho} = \textbf{correlation matrix}$$

So **when you do PCA on the standardized dataset**, you don't take the **eigenvectors of** Σ but of $V^{-1/2}\,\Sigma\,V^{-1/2}$ which is **the correlation matrix**, NOT THE COVARIANCE MATRIX.

$$\rho = \sum_{i=1}^{p} \lambda_i \, \underline{e_i} \underline{e_i}^T$$

$$y = P^T Z = P^T V^{-1/2}(X - \mu)$$

P^T *is the matrix coming out from the spectral decomposition* ***of the correlation***

$$\rho = P \Lambda P^T$$

$tr(\rho) = tr(\Lambda) = p$ *the total variance one you standardize is the amount of features*

The average value of the eigenvalues is 1 (since is standardized):

$$\frac{\sum_j^N \lambda_j}{p} = 1$$

So, rule to select components:

- **Select components whose eigenvalue larger than 1** (larger than the mean).

If you do regression, is the same for standardized and not standardized. Because linear regression is scale invariant.
What is really done is not to get eigenvalues of Σ, since μ *and* Σ are unknown.

$$\Sigma \text{ is estimated by } S$$
$$\mu \textit{ is estimated by } \overline{X}$$

$$S = \sum_{i}^{p} \lambda_i \underline{e_i} \underline{e_i}^T = P \Lambda P'$$

And again, $Y = XP$.
Y is the new dataset, projected on the principal components.
Dimensionality reduction by taking the first components that explain more variability.

But also the smallest eigenvalues can be interesting**: if is $\lambda_p = 0$, or close to 0 => there is a linear relationship between components**, so, data don't live in R^p but in a smaller subspace.

$$\text{Var}(y_p) = 0$$

You get component that is collinear to other, perfectly explaining the same thing of another component.

A geometrical perspective of PCA

Finding the linear subspace (smaller than the original one) closest to the data.
Is **NOT** the regression line.
In regression you minimize the residual from the line.
In PCA you take the perpendicular to the line and you take the distance from the point to the line.
PCA is independent on the reference system.

PCA wants to find orthonormal basis $\boldsymbol{\eta_1}, \ldots, \boldsymbol{\eta_k}$ that generate a space the closest as possible to the data cloud.

The projection of the centred data into the new space is:

$$\pi_{x_i \mid span(\eta_1,\ldots,\eta_k)} = \sum_{j=1}^{k} \eta_i \eta_j^T (x_i - \underline{\bar{x}})$$

The orthonormal basis to find is $\eta_1, \ldots, \eta_k$ such that the distance from the point and its projection is minimal:

$$\sum_{i}^{n} ||(x_i - \underline{\bar{x}}) - \sum_{j=1}^{k} \eta_i \eta_j^T (x_i - \underline{\bar{x}})||^2 \qquad is\ minimal$$

If you decompose the square you end up with Pitagora:

$$\underbrace{\sum_{i}^{n} (x_i - \underline{\bar{x}})^T (x_i - \underline{\bar{x}})}_{\text{This don't depend on basis}} - \underbrace{\sum_{i=1}^{n} \sum_{j=1}^{k} (x_i - \underline{\bar{x}})^T \eta_i \eta_j^T (x_i - \underline{\bar{x}})}_{\text{This is what you can maximize by choosing the basis}} \quad is\ minimal$$

$$\sum_{i=1}^{n} \sum_{j=1}^{k} (x_i - \underline{\bar{x}})^T \eta_i \eta_j^T (x_i - \underline{\bar{x}}) = (n-1) \sum_{j=1}^{k} \eta_j^T S\, \eta_j$$

For k = 1, the η that maximizes that object is the eigenvector of the spectral decomposition of S (for the lemma of the quadratic functions): $\eta_1 = e_1$.
By induction, $\eta_1 = e_1, \eta_2 = e_2, \ldots, \eta_k = e_k$.

The linear space that minimizes the distance from the data is the one identified by the k first eigenvector of S.

Define the **approximation error / Sum of squared residuals** (what you leave out by not considering the last *p - k* components):

$$\sum_{i}^{n} ||(x_i - \underline{\bar{x}}) - \sum_{j=1}^{k} \eta_i \eta_j^T (x_i - \underline{\bar{x}})||^2 = (n-1) \sum_{j=k+1}^{p} \lambda_j$$

7 Mar

Is the mean of a population different from the mean of another population? Not asking about the $\underline{\bar{x}}$, but about the $\underline{\mu}$.
Same for the covariance and sample covariance.

We need a model for this problem.

Gaussian model

For any p.
The density is:

$$\emptyset(\underline{t}) = \frac{1}{\sqrt{(2\pi)^p Det(\Sigma)}} \exp\left[-\frac{1}{2}\left(\underline{t}-\underline{\mu}\right)^T \Sigma^{-1}\left(\underline{t}-\underline{\mu}\right)\right]$$

Square of the Mahanalobis distance at the exponent.
The density is a decreasing exponential starting from μ: the more you get far from μ, in terms of Mahanalobis distance, the more it decays.
For $\Sigma = I$, you go in Euclidean distance.

The density gives an ellipse induced by the eigenvectors of Σ. Radius proportional to $\sqrt{\lambda}$.

Theorem

$$\underline{X} \sim N_p(\mu, \Sigma) \quad \textbf{\textit{IFF}} \quad \forall\, a \in R^p, \quad a^T\underline{X} \sim N_1(a^T\mu, a^T\Sigma a)$$

For each vector in p dimension, on that linear combination must be normal: is impossible to prove.

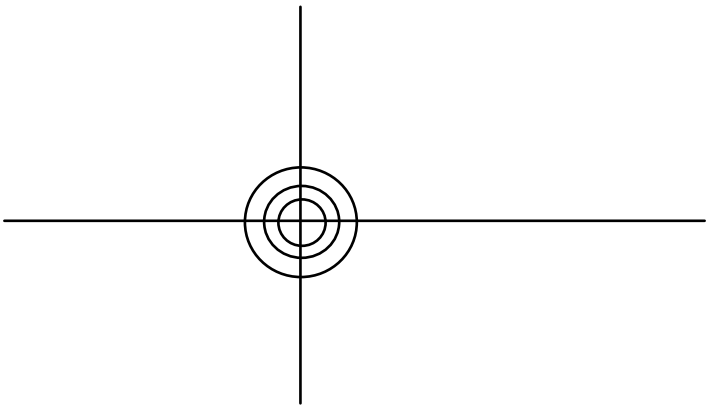
There is a simpler corollary, only going into a single direction.

Corollary

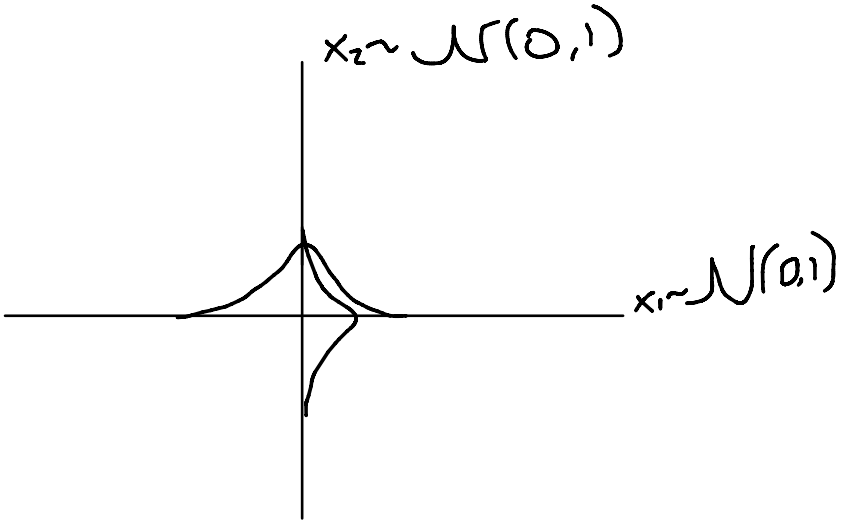
$$if\ \underline{X} \sim N_p(\mu, \Sigma) \quad \textbf{\textit{then}} \quad X_i \sim N_i(\mu_i, \sigma_{ii}), \quad i = 1, \ldots, p$$

This is saying: if a vector is gaussian, then all its components are gaussian.

Note: if all components are gaussian, it doesn't say the vector is gaussian.
But we use this because if a component is not gaussian, then for sure won't be multivariate gaussian.
Example:
For $X \sim N_2(0, \mathrm{I})$, contour plots are circles centred in zero with all same distance.



The distribution of x1 and x2 is gaussian , so:



By integrating this, you get a semicircle in first quadrant and in third quadrant: either both positive or both negative.

If you take the marginal distribution (integrating the gaussians, the one horizontal and the vertical one), you get again the same distribution with circles, that is gaussian. However the multivariate distribution is not.

Gaussianity doesn't exists. Some data fits to gaussian model.
You need to transform your data to make it gaussian, in order to exploit the theory of gaussian distributions.
Use always non-linear transformations.

Proposition for linear combination of gaussian distributions
Linear transformations of gaussian distributions is gaussian.

$$Given\ \underline{X} \sim N_p\left(\underline{\mu}, \Sigma\right), \quad and \quad A \in R^{qxp}$$

Then

$$A\underline{X} \sim N_q(A\,\underline{\mu}, A\,\Sigma\,A^{\mathrm{T}})$$

Proposition
Translating a random vector only translate its mean

$$Given\ \underline{X} \sim N_p\left(\underline{\mu}, \Sigma\right), \quad and \quad \underline{d} \in R^p$$

Then

$$\underline{X} + \underline{d} \sim N_p(\underline{\mu} + \underline{d}, \Sigma)$$

- ➔ If the components are gaussian and independent, then the distribution is multivariate gaussian distribution. (for this reason we check at the principal component because they are uncorrelated (not independent, they'd be independent if the distribution is gaussian. See later for uncorrelation and independence in gaussian world)).

How to generate a multivariate gaussian distribution

From collection of iid gaussian distributions univariate to multivariate gaussian (in p dimension), with zero mean and variance/covariance as the identity:

- generate p **independent** univariate standard normal and put them together
  - $\underline{Z} \sim N_p(\underline{0}, I)$
- Take a linear combination of this vector (using the propositions of before):

$$\Sigma^{1/2}\underline{Z} + \underline{\mu} \sim N_p\left(\underline{\mu}, \Sigma^{1/2} I \Sigma^{1/2}\right) = N_p\left(\underline{\mu}, \Sigma\right)$$

How to get a set of gaussian variables from a multivariate distribution

You can go the other way around like this:

- $from\ \underline{X} \sim N_p\left(\underline{\mu}, \Sigma\right)$
- $\Sigma^{-\frac{1}{2}}(\underline{X} - \underline{\mu}) \sim N_p(\underline{0}, I)$

Chi square, what is it? Distribution of radius of Ellipse.
A chi square of degree p is the sum of p squared normal distributions.

$$\underline{X} \sim N_p\left(\underline{\mu}, \Sigma\right)$$

$$d_{\Sigma^{-1}}\left(\underline{X}, \underline{\mu}\right) = \left(\underline{X} - \underline{\mu}\right)^T \Sigma^{-1} \left(\underline{X} - \underline{\mu}\right)$$

$$= \left(\underline{X} - \underline{\mu}\right)^T \Sigma^{-1/2} \underbrace{\Sigma^{-1/2}\left(\underline{X} - \underline{\mu}\right)}_{\underline{Z} \sim N(0,1)} = \underline{Z}^2 = \Sigma_{i=1}^{p} \underline{Z_i^2} \sim \chi^2(\mathrm{p})$$

For how is this distribution, is useful to say something about the true mean and the estimate.
With probability $1 - \alpha$ I fall into this ellipse (with radius proportional to the eigenvectors of sigma):
$\Pr\left[\left(\underline{X} - \underline{\mu}\right)^T \Sigma^{-1} \leq \chi_{1-\alpha}{}^2(\mathrm{p})\right] = 1 - \alpha$
If we take the Mahanalobis distance, we get an ellipse centred in μ with squared radius of $\chi_{1-\alpha}{}^2(\mathrm{p})$ that contains $1 - \alpha$ percent of the mass.
I can do the same reasoning with the point I'm observing: centring the ellipse in $\overline{\underline{x}}$ and see if $\underline{\mu}$ is inside the ellipse. (distances are symmetric).
If the distance (in terms of Mahanalobis) from $\underline{\mu}$ to $\overline{\underline{x}}$ is smaller than 95%, also the distance between $\overline{\underline{x}}$ and $\underline{\mu}$ is smaller than 95%.

➔ **Confidence interval for the mean.**

Notation:

- Two blocks of features $\underline{X_1}, \underline{X_2}$. So vector $\underline{X} \in R^p$, with $\underline{X} = (\underline{X_1}, \underline{X_2}), with\ \underline{X_1} \in R^q, \underline{X_2} \in R^{p-q}$.
- $\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$

Proposition: subset of components of $\underline{X}$ $(\sim N_p(\underline{\mu}, \Sigma))$ is still gaussian

➔ $\underline{X_1} \sim N_q(\underline{\mu_1}, \Sigma_{11})$
➔ $\underline{X_2} \sim N_{p-q}(\underline{\mu_2}, \Sigma_{22})$

Σ_{12} *and* Σ_{21} contain the cross covariances between different features. Say in 1 there is weight, height, colour hair and in 2 there is salary and wealth, then Σ_{12} *and* Σ_{21}contains the covariances between those features crossed.

Proposition
Zero covariance means independent **for gaussian distribution. (uncorrelation = independence)**
If $\Sigma_{12} = 0$

➔ $\underline{X_1}$ *is stocastically independent from* $\underline{X_2}$

For general distribution, instead, is a sufficient condition but not necessary. If two things are independent, the covariance can be zero.
In general distributions: if covariance is zero is not said that they are independent.

Theorem: how to do inference about distribution

If I know the distribution of $\underline{X_2}$ (say salary and wealth) and I want to get the distribution of $\underline{X_1}$ (the height, weight and hair colour) it means I want the distribution of $\underline{X_1}$ conditioned on the value of $\underline{X_2}$.

$$\underline{X_1} \mid \underline{X_2} \sim N_q(\underline{\mu_1} + \Sigma_{12}\Sigma_{22}^{-1}\left(\underline{x_2} - \underline{\mu_2}\right), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$$

Note that, if you don't know anything about $\underline{X_2}$, $\underline{X_1}$ is $\sim N_q\left(\underline{\mu_1}, \Sigma_{11}\right)$. But having some information, you have the distribution above. So with the correlation you're gaining information about the other variable.

Indeed, if $\underline{X_2}$ and $\underline{X_1}$ are independent, meaning $\Sigma_{12} = 0$, you end up having again $\underline{X_1} \sim N_q\left(\underline{\mu_1}, \Sigma_{11}\right)$.

➔ **We need dependency between features.**
- That's what allows us to make inference. We want **independent** random **variables** (iid)

Partial Covariance
$\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$: this is what decrease the variability of what I want to predict. Can be computed before getting the observation of $\underline{X_2}$.
If there is no covariance between the predictors and what I want to predict, every ML model or any powerful model is useless.

A comfortable example:

$$Let\ p = 2, \quad so\ \underline{X_1} = x, \underline{X_2} = y$$

And $\Sigma_{11} = \sigma_{xx}, \Sigma_{22} = \sigma_{yy}, \Sigma_{12} = \sigma_{xy}$.

$$y \sim N_1(\mu_y, \sigma_{yy})$$

The uncertainty about our target variable can be seen as this confidence interval:

$$P\left[y \in \left[\mu_y \pm 2\sqrt{\sigma_{yy}}\right]\right] = 0.95$$

Now, you have also x, therefore for the theorem you can say:

$$y \mid x \sim N_q(\mu_y + \sigma_{xy}\sigma_{xx}^{-1}(x - \mu_x), \sigma_{yy} - \sigma_{xy}^2\sigma_{xx}^{-1})$$

The correlation is

$$\rho = \frac{\sigma_{xy}}{\sqrt{\sigma_{xx}\sigma_{yy}}}$$

So we can rewrite the distribution as:

$$y \mid x \sim N_q(\mu_y + \frac{\sigma_{xy}}{\sigma_{xx}}(x - \mu_x), \sigma_{yy}(1 - \rho^2))$$

The higher the correlation ($\rho \cong 1$) the more you reduce the variability.

And the mean of y given x is the **regression**:

$$E[y \mid x] = \mu_y + \frac{\sigma_{xy}}{\sigma_{xx}}(x - \mu_x)$$

➔ **In the gaussian world, the regression function** (the expected value of the target given the features you know ($E[y \mid x]$), hence the best we can do to minimize the MSE) **is a line**

The formula for regression line:

$$\frac{y - \mu_y}{\sqrt{\sigma_{yy}}} = \rho \frac{x - \mu_x}{\sqrt{\sigma_{xx}}}$$

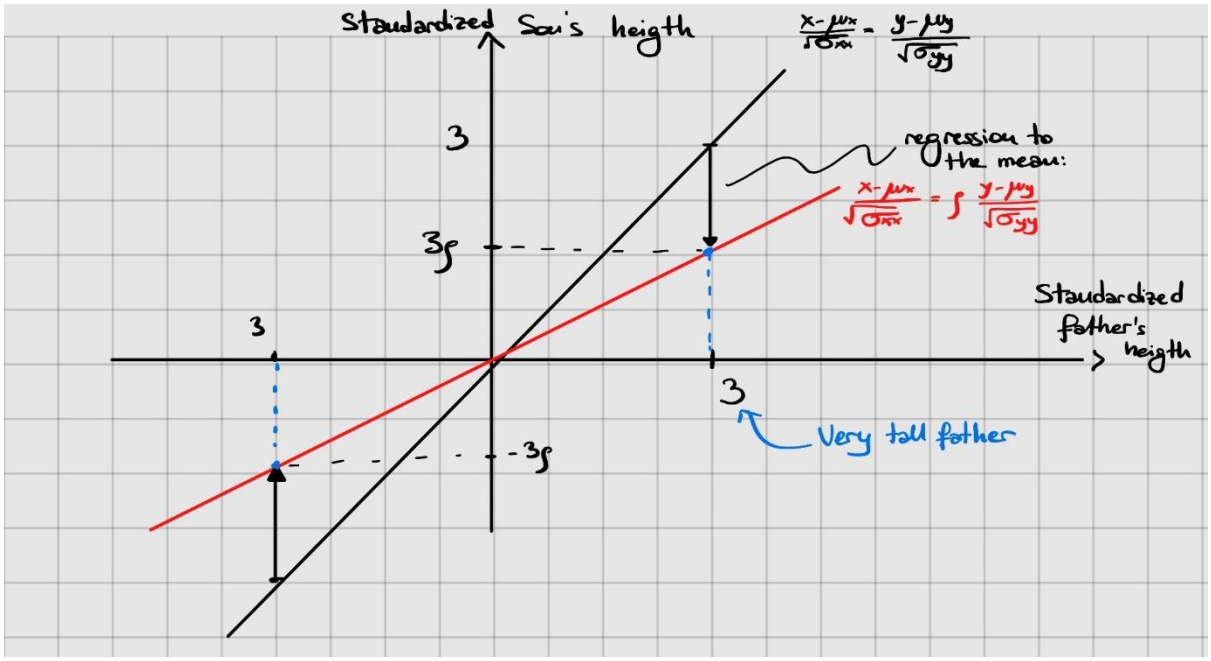
- This is telling that, the link between the standardized target and the standardized feature is a line going through the origin and **slope of ρ**

The regression to the mean: if you take a very high father (in terms of variability), his son will be if there shorter than him (**in terms of son's variability**), because of the correlation between father's height and son's height is not 1. The exceptionalities are lost in the regression, since you're regretting to the mean. See on the plot how you go closer to the mean (the x-axis) both for very high values and low values.
Very short fathers will have not so short son in the son's populations.

That's it. Is wrong the **regression fallacy:** there is no cause-effect. If my average grades are 30, and in the next exam I take 24 is not that is because I haven't studied.

When you have repeated measures, you tend to have worst or better measurements (depending on where you are) and you interpret as "because something happened". **NO. Is because of uncertainty.**

Again examples in economics: companies that at the beginning are bad, then they become good, and the opposite.



8-3

Random vectors are realized by observations of vectors in R^p.
The units that are observed are chosen to be independent. The independency is hard to obtain in real world.

We estimate (μ, Σ) with sample mean $\overline{x}$ and sample covariance S which are unbiased: on average they give the right $(\mu \; and \; \Sigma)$.

But more, under the assumption of Gaussianity and independency (iid) they are also MLE.

Flip coins randomly: you get 3 heads and 2 tails. The sample average is 3/5.
Can we justify this result? What is the probability of observing 3 heads and 2 tails? $\hat{p} = \frac{3}{5}$
$P[x_1 = 1, x_2 = 1, x_3 = 0, x_4 = 0, x_5 = 1] = p^3(1-p)^2 \qquad if \; we \; assume \; Bernoulli \; and \; independency$

Likelihood function: how likelihood is to observe that data (3 heads and 2 tails)? Is general, for every p. **Depends on the observation data.**

Fisher's idea: let's find the value of p that maximize the likelihood of what we've observed.

Story time

Sherlock: observe data of the crime. Woman on the floor with a knife on the chest. Blood around the room. With husband near to the woman, full of blood, with a knife on his hand. That's the data.

Values of parameters: story of the husband: "we're in the kitchen preparing the dinner, the wife is with knife. The asteroid dump on the knife of the woman, and goes into the chest." <- value 1 of parameter.
Second value of parameter: "you killed the woman with your knife".

Which value is more likely? Value 2.

Agree to the hypothesis that maximize the likelihood.

MLE definition: choose the parameter (in Bernoulli the *p*) of the distribution that maximize the function likelihood.

Bernoulli case:

$$L(p \mid x_1, \dots, x_n) \;=\; p^{\sum x_i}(1-p)^{n-\sum x_i}$$

MLE:

$$\boldsymbol{argmax}_{\boldsymbol{p}\in[\mathbf{0},\mathbf{1}]}\, L(p) = \frac{1}{n}\sum x_i$$

- **Invariance of MLE:** $let\ \vartheta\ \in R^k\ be\ the\ parameters. The\ data: x_1, \dots, x_n.$
  - **In Gaussian world, unbiased estimators are also MLE estimators.**
    - This mean: let h be a function from $R^k \rightarrow R^j$.
      If $\hat{\vartheta}$ is the MLE of ϑ, then $h(\hat{\vartheta})$ is the MLE of $h(\vartheta)$

MLE for gaussian distribution.

$$L\left[\underline{\mu}, \Sigma \;\middle|\; X_1, \dots, X_n\right] = \prod \frac{1}{\sqrt{(2\pi)^p Det(\Sigma)}} \exp\left[-\frac{1}{2}\left(x_i - \underline{\mu}\right)^T \Sigma^{-1}\left(x_i - \underline{\mu}\right)\right]$$

$$l\left(\underline{\mu}, \Sigma \;\middle|\; X_1, \dots, X_n\right) = nlog\frac{1}{\sqrt{(2\pi)^p Det(\Sigma)}} - \frac{1}{2}\sum\left(x_i - \underline{\mu}\right)^T \Sigma^{-1}\left(x_i - \underline{\mu}\right)$$

$$\hat{\underline{\mu}}, \hat{\Sigma} = \text{argmax}_{\underline{\mu},\underline{\Sigma}}\, l(\mu, \Sigma \mid X_1, \dots, X_n)$$

You can summarize the whole data with only $\hat{\mu}, \hat{\Sigma}$ they are **sufficient statistics.** "Data compression".

What distribution does $\underline{\overline{X}}, \hat{\Sigma}, and\ S$ follow?

- **Distribution of $\underline{\overline{X}}$**: If $\underline{x}_1, \dots, \underline{x}_n\ are\ iid \sim N_p\left(\underline{\mu}, \Sigma\right) => \underline{\overline{X}} \sim N_p\left(\underline{\mu}, \frac{1}{n}\Sigma\right)$
  - If random vectors are gaussian, the sample mean is gaussian with covariance rescaled
  - Note: is not the large law number, can have limited, independent gaussian sample size
    The LLN is saying: n very big and independent sample size (with any distribution)
  - If n is very big, you have lower variance, because of 1/n term.
- **Distribution of $\hat{\Sigma}$: distribution of random matrix.**

➔ We need a new distribution: **Wishart.**

Wishart distribution

$Let\ \underline{Z}_1, \dots, \underline{Z}_m\ iid \sim N_p(\underline{0}, \Sigma)$.

➔ $\sum_{i=1}^{m} \underline{Z}_i\, \underline{Z}_i^T \sim Wish(\Sigma, \mathrm{m})$

How to remember: the chi square with degree p was a sum of p squared normal distributions. Wishart is for matrix. Wish with a covariance and m degree, is a sum of m matrix built from gaussian distributions ($\underline{Z}_i\, \underline{Z}_i^T$). Is a sort of matrix version (or multivariate extension) of chi square, apart from a the covariance term.

Properties:

- Given two Wishart distributions $A_1 \sim Wish(\Sigma, \mathrm{m}_1), A_2 \sim Wish(\Sigma, \mathrm{m}_2)$
  - If they are independent:
    - $A_1 + A_2 \sim Wish(\Sigma,\ \mathrm{m}_1 + m_2)$
- If $C \in R^{kxp}$ matrix and $\mathrm{A} \sim Wish(\Sigma, \mathrm{m})$
  - $C\ \mathrm{A}\ \mathrm{C}^{\mathrm{T}} \sim Wish\left(\mathrm{C}\ \Sigma\ \mathrm{C}^{\mathrm{T}}, \mathrm{m}\right)$
- $Let\ a\ \in R, with\ a > 0$
  - $aA \sim Wish(a\Sigma, \mathrm{m})$
- Is a multivariate extension of the χ^2.
  - If $p\ =\ 1, \Sigma$ is a number, let's call it σ^2. So, $A \sim Wish(\sigma^2, \mathrm{m})$.
    This means that $Z_1, \dots, Z_m\ iid \sim N_1(0, \sigma^2)$.

$$A = \sum_{i=1}^{m} Z_i\, Z_i^T = \sum_{i=1}^{m} Z_i^2$$

$$\frac{A}{\sigma^2} = \sum_{i=1}^{m} \frac{Z_i^2}{\sigma^2} \sim \chi^2(\mathrm{m}) \ => A \sim \sigma^2\, \chi^2(\mathrm{m})$$

Remark: is the random variable that you divide by σ^2, not the distribution that you multiply. The latter is just a notation.

- **Distribution of S**: remembering that $\underline{x}_1, \dots, \underline{x}_n\ are\ iid \sim N_p\left(\underline{\mu}, \Sigma\right)$
  - $(n-1)S = \sum_{i=1}^{n}\left(\underline{x}_i - \underline{\overline{x}}\right)\left(\underline{x}_i - \underline{\overline{x}}\right)^T \sim Wish(\Sigma, \mathrm{n}-1)$
    - ***Note***: removed one degree of freedom because we have $\underline{\overline{x}}$ in the estimator, and not $\underline{\mu}$. If we'd $\underline{\mu}$, we'd have Wishart with n, and not $\mathrm{n}-1$.

$\underline{\overline{X}}$ ***and*** $\boldsymbol{S}$ ***(and*** $\underline{\overline{X}}$ ***and*** $\hat{\Sigma}$***) are stochastically independent*****: knowing one doesn't give me anything for the other.**

- $S \sim Wish\left(\frac{\Sigma}{\mathrm{n}-1}, \mathrm{n}-1\right)$
- $\hat{\Sigma} = \frac{n-1}{n} S \sim Wish\left(\frac{\Sigma}{\mathrm{n}}, \mathrm{n}-1\right)$

All of what we've said is true for every n. Big or small. We "only" need independent identical distributed n gaussian variables.
But what can we do with large samples with not Gaussianity?

Central limit theorem:

Given a sequence of random vectors $\underline{x}_1, \dots, \underline{x}_n$ ***i.i.d.*** **with any distribution** with $E\left[\underline{x}_i\right] = \underline{\mu}_i$ and $Cov\left(\underline{x}_i\right) = \Sigma$.

➔ $\underline{\overline{X}} \sim N_p\left(\underline{\mu}, \frac{1}{n}\Sigma\right)$

- *The sample mean has distribution that can be approximated by a gaussian*

"CLT is for distribution of the expected value"

Large law number

Given a sequence of random vectors $\underline{x}_1, \dots, \underline{x}_n$ ***i.i.d.*** **with any distribution** with $E\left[\underline{x}_i\right] = \underline{\mu}_i$ and $Cov\left(\underline{x}_i\right) = \Sigma$.

➔ $\underline{\overline{X}} = \frac{1}{n}\sum_{i=1}^{n} \underline{x}_i$ *converges to* $\underline{\mu}$ *as* $n \to \infty$

➔ $S = \frac{1}{n-1}\sum_{i=1}^{n}\left(\underline{x}_i - \underline{\overline{x}}\right)\left(\underline{x}_i - \underline{\overline{x}}\right)^T$ *converges to* Σ *as* $n \to \infty$

"LLN is for convergence of estimate to the true value"

11-3

Inference for the mean $\mu \in R^p$, if we have n very large (n >> p)

Sample $\underline{x}_1, \dots, \underline{x}_n$ ***i.i.d.*** **with any distribution,** with some μ *and* Σ.

Since n is very large

➔ for the CLT we can approximate the distribution of the sample mean $\underline{\overline{X}} \sim N_p\left(\underline{\mu}, \frac{1}{n}\Sigma\right)$

- therefore, I know the distribution of the Mahanalobis distance between $\underline{\overline{X}}$ and $\underline{\mu}$:

$$\left(\underline{\overline{X}} - \underline{\mu}\right)^T \left(\frac{1}{n}\Sigma\right)^{-1} \left(\underline{\overline{X}} - \underline{\mu}\right) \sim \chi^2(\mathrm{p})$$

$$\underbrace{n\left(\underline{\overline{X}} - \underline{\mu}\right)^T \Sigma^{-1} \left(\underline{\overline{X}} - \underline{\mu}\right)}_{*} \sim \chi^2(\mathrm{p})$$

If Σ is known, the random quantity * (distance) depends on the true mean $\underline{\mu}$, but its distribution doesn't depend on it. This is called a **Pivotal quantity.**

That pivotal quantity can't be used since we don't know Σ.
But, remember that we assumed n very large, so we can use LLN:
$S \to \Sigma$, therefore:

$$n\left(\underline{\overline{X}}-\underline{\mu}\right)^T S^{-1}\left(\underline{\overline{X}}-\underline{\mu}\right)\sim \chi^2(\mathrm{p})$$

How to use pivotal quantity

Fixing an α, I want to see the probability that my pivotal (random) quantity is less than a quantile.

$$\underbrace{\Pr\left[n\left(\underline{\overline{X}}-\underline{\mu}\right)^T S^{-1}\left(\underline{\overline{X}}-\underline{\mu}\right)\right.}_{d^2_{(\frac{1}{n}S)^{-1}}}\leq \left.\chi^2_{1-\alpha}(\mathrm{p})\right]=1-\alpha \qquad (*)$$

Remark: $1-\alpha$ is the probability to fall on the left of the chi square.
($\chi^2_{1-\alpha}(\mathrm{p})$ is the point on the x axis that divide $1-\alpha$ and α.)

We define two ellipses, the first centred in $\underline{\mu}$ and the other in $\underline{\overline{X}}$

- $E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\mu}\right)=\{\underline{x}\in R^p: d^2_{\left(\frac{1}{n}S\right)^{-1}}(\underline{x},\underline{\mu})\leq \chi^2_{1-\alpha}(\mathrm{p})\}$
- $E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\overline{X}}\right)=\{\underline{x}\in R^p: d^2_{\left(\frac{1}{n}S\right)^{-1}}(\underline{x},\underline{\overline{X}})\leq \chi^2_{1-\alpha}(\mathrm{p})\}$

They have same axis, same length of semiaxis but different centre.
Since distance is symmetric:

$$\underline{\overline{X}}\in E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\mu}\right) \textbf{\textit{IFF}}\ \underline{\mu}\in E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\overline{X}}\right)$$

Therefore, we can rewrite the equation (*):

$$Pr\left[\underline{\overline{X}}\in E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\mu}\right)\right]=1-\alpha$$

And equivalently:

$$Pr\left[\underline{\mu}\in E^{\alpha}_{\chi^2_{1-\alpha}(p)}\left(\underline{\overline{X}}\right)\right]=1-\alpha$$

The only thing that is random is $\underline{\overline{X}}$, so, the ellipse and its form. But $\underline{\mu}$ is fixed, is a parameter.

You give the observations, and you compute $\underline{\overline{X}}$, so you have the ellipse. This ellipse with probability $1-\alpha$ will cover $\underline{\mu}$.

➔ **Confidence region for $\underline{\mu}$ at level $1-\alpha$**

$$CR_{1-\alpha}\left(\underline{\mu}\right)=\{\underline{\eta}\in R^p: n\left(\underline{\eta}-\underline{\overline{X}}\right)^T S^{-1}\left(\underline{\eta}-\underline{\overline{X}}\right)\leq \chi^2_{1-\alpha}(p)\}$$

Ellipse **centred in $\underline{\overline{X}}$,** with radius given by the quantile of chi distribution, and axes are the eigenvector of S.

Once you have data, you can compute this confidence region for $\underline{\mu}$.

Note: you can't say $\underline{\mu}$ belongs to this CR with 95% probability.

Because there is no randomness, no probability. The ellipse is realized by the data.

➔ Once you have the ellipse, $\underline{\mu}$ is either inside or outside. With no probability! But of course you don't know where it is since you don't know $\underline{\mu}$.

It is called **confidence region** for a reason and not probability region: because you produce an ellipse with an algorithm that 95% of the times does the right thing.

Is not that 95% of the time is inside and 5% is outside.

We can't talk about probability of parameters. Parameters are not random, they exists fixed, but we don't know them.

Test

Sometimes you don't want to estimate the parameters, but to test an hypothesis on the parameter. You always assume H_0 is true, and you want to prove is false.

$$H_0: \underline{\mu} = \underline{\mu}_0 \qquad vs \qquad H_1: \underline{\mu} \neq \underline{\mu}_0$$

The old drug has to be worst respect to the new drug. So we have to prove is different.

The philosophy of testing is always assuming that H_0 is true, and so that there is a quantity (T_0^2) that follow a distribution. You reject H_0 if with data you end in a quantity of T_0^2 that is really unlike to happen.

Use data to prove H_0 is false: $\overline{\boldsymbol{X}}$ must be far from $\underline{\mu}_0$.

➔ Far in the sense of this distance, called **T square statistic:**

- $T_0^2 = n\left(\overline{\underline{X}} - \underline{\mu}_0\right)^T S^{-1}\left(\overline{\underline{X}} - \underline{\mu}_0\right)$ used to test the mean of a multivariate distribution

  If is large, you can reject H_0. But how large?
  We know that **if $\boldsymbol{H_0}$ *is true* $\rightarrow \boldsymbol{T_0^2} \sim \boldsymbol{\chi^2}(\mathbf{p})$.**
  So you can fix a threshold on this distribution, in the point on the right (on x axis) where $\alpha\%$ of the situation where H_0 *is true* is rejected by the test. You are concluding wrong only $\alpha\%$ of the time with this policy:

  $$reject\ if\ \{T_0^2 > \chi^2_{1-\alpha}(\mathrm{p})\}$$

  Another reasoning you can follow is to not fix α, but compute only T_0^2, and the area on the right of what you've observed is the **p-value.** In this way, you can reject at any probability level higher than the p-value. (you want small p-value to reject H_0).

  $$\boldsymbol{p - value} \leq \alpha \qquad \boldsymbol{IFF} \qquad T_0^2 > \chi^2_{1-\alpha}(\mathrm{p})$$

  Reasoning with Confidence Regions (same thing with different perspective)**:** you reject if

  $T_0^2 = n\left(\overline{\underline{X}} - \underline{\mu}_0\right)^T S^{-1}\left(\overline{\underline{X}} - \underline{\mu}_0\right) > \chi^2_{1-\alpha}(\mathrm{p})$

  This happen **IFF**:

  $$\underline{\mu}_0 \notin \mathrm{CR}_{1-\alpha}(\underline{\mu})$$

  This mean: reject if $\underline{\mu}_0$ doesn't belong to the confidence region at level α for $\underline{\mu}$.

The confidence region for $\underline{\mu}$ identifies all the values of the mean for which you can't reject H_0.

---------- Until here valid only if n is large ----------

But what happens if n is not very large?

I need to specify the distribution of my samples: we have to assume Gaussianity.

$$\underline{X}_1, \dots, \underline{X}_n \; iid \sim N_p\left(\underline{\mu}, \Sigma\right)$$

(transform the data to make this assumption reasonable. Data don't follow gaussian distribution)
So, having that distribution, we have that:

$$n\left(\overline{\underline{X}} - \underline{\mu}\right)^T S^{-1}\left(\overline{\underline{X}} - \underline{\mu}\right) \sim F$$

Fisher distribution

Having two independent chi square distributions, $y \sim \chi^2(n), w \sim \chi^2(\mathrm{m})$
The F distribution is the distribution of their ratio.

$$\frac{y/n}{w/m} \sim F(n, m)$$

It's a **generalization of the t-distribution:**

$t \sim t(m), \quad with\ t = \frac{Z}{\sqrt{w/m}}$ where $Z \sim N(0,1), w \sim \chi^2(\mathrm{m})\ and\ z\ independent\ from\ w$

$$t^2 = \frac{Z^2}{w/m} \sim F(1, m)$$

Note: $for\ m \to \infty, F(n,m) \to \frac{1}{n}\chi^2(\mathrm{n})$. (Since summing square gaussian is the variance, for the denominator you have m very large so LLN, and variance is 1).

Hotelling theorem

We want to see the distribution of a distance induced by a generic random matrix, from a gaussian vector and its mean.
Assume:

- $\underline{X} \sim N_p\left(\underline{\mu}, \Sigma\right), \quad with \det(\Sigma) > 0$
- $mW \sim Whis(\Sigma, m)$
- $\underline{X}\ and\ W\ independent$

➔ $\frac{m-p+1}{mp}\left(\underline{X} - \underline{\mu}\right)^T W^{-1}\left(\underline{X} - \underline{\mu}\right) \sim F(p, m-p+1)$

We're computing the distance between a gaussian random vector and its mean when the geometry is not generated by Σ, but is generated randomly by W.
If it was generated by Σ the distance (Mahanalobis) would have followed a chi-square distribution.

How can we use it? With a corollary:

$$\underbrace{n\left(\overline{\underline{X}} - \underline{\mu}\right)^T S^{-1}\left(\overline{\underline{X}} - \underline{\mu}\right)}_{d^2_{S^{-1}}(\overline{\underline{X}}, \underline{\mu})} \sim \frac{(n-1)p}{n-p} F(p, n-p)$$

➔ The Mahanalobis distance has an F distribution (not anymore the chi square as the case with large n).

Proof: S has Wishart distribution. Sample mean $\overline{\underline{X}}$ and sample covariance S are independent.

Hotelling's T^2 statistic

$$n\left(\underline{\overline{X}}-\underline{\mu}\right)^T S^{-1}\left(\underline{\overline{X}}-\underline{\mu}\right)$$

Is a pivotal quantity: depends on $\underline{\mu}$ but the distribution does not.

➔ I can use it to build confidence regions and tests

$$CR_{1-\alpha}\left(\underline{\mu}\right)=E^{\alpha}_{S^{-1}}(\underline{\overline{X}})=\{\underline{\eta}\in R^p: n\left(\underline{\overline{X}}-\underline{\mu}\right)^T S^{-1}\left(\underline{\overline{X}}-\underline{\mu}\right)\leq \underbrace{\frac{(n-1)p}{n-p}F_{1-\alpha}(p,n-p)}_{radius^2}\}$$

➔ For n large, the $radius^2=\chi^2_{1-\alpha}(\mathrm{p})$

➔ For n small, with Gaussianity, the $radius^2=\frac{(n-1)p}{n-p}F_{1-\alpha}(p,n-p)$

But what happens if n is large, **and** we have Gaussianity? The two points above coincide (of course):

$$n\to\infty,\qquad \frac{(n-1)p}{n-p}\to 1$$

$$n\to\infty,\qquad F(p,n-p)\to\frac{1}{p}\chi^2_{1-\alpha}(\mathrm{p})$$

For big n, $\chi^2(n)\to 1,\ that\ means\ no\ variance\ for\ big\ sample.$

Test with small n

$$H_0:\underline{\mu}=\underline{\mu}_0\qquad vs\qquad H_1:\underline{\mu}\neq\underline{\mu}_0$$

If H_0 is true $T_0^2=n\left(\underline{\overline{X}}-\underline{\mu}_0\right)^T S^{-1}\left(\underline{\overline{X}}-\underline{\mu}_0\right)\sim\frac{(n-1)p}{n-p}F(p,n-p)$. (And not a chi square as before)

➔ Reject H_0 if $T_0^2\geq\frac{(n-1)p}{n-p}F(p,n-p)$

P-value: we must compute the area on the right of the distribution: you have to multiply $\frac{n-p}{(n-1)p}$ before T_0^2.

Again, $CR_{1-\alpha}\left(\underline{\mu}\right)$ is the set of $\underline{\mu}_0$ that you can't reject at level α.

Example: if you have $S=I$ the confidence interval for the mean is a circle: same range of value for the two components. Same uncertainty for the two components.
If you have higher off-diagonal components, the ellipse is squishing because the correlation between components is reducing the uncertainty between them.
In both cases the mean is the same, without changing the point to estimate $\underline{\overline{X}}$, but only changing the covariance

12-3

Inference for linear combination of $\underline{\mu}$

Given $\underline{x}_1, \dots, \underline{x}_n \sim N_p\left(\underline{\mu}, \Sigma\right)$
I want to estimate the mean of $\underline{a}^T\underline{\mu}$, that is a number. $\underline{a}^T\underline{\mu} \in R$.

➔ Stat101 case.

$\underline{a}$ can be very general, if it is with only a 1, it's used to calculate the mean of a single feature.
If it is like $\underline{a} = (0\ 0\ 0\ 1\ 0\ 0\ -1)$ we want to make comparison: if $\underline{a}^T\underline{\mu} \geq 0$ it means there is a change (say one is the salary at the start of the carrier, and one of the end. We want to see if there were an increase).

Estimator for $\underline{a}^T\underline{\mu} = \underline{a}^T\overline{\underline{X}}$. For the invariance property of MLE, it's a MLE too.
But I want to see also the uncertainty of this estimate, so, use confidence intervals.

$$\underline{a}^T\overline{\underline{X}} \sim N_1(\underline{a}^T\underline{\mu}, \frac{1}{n}\underline{a}^T\Sigma\underline{a})$$

And therefore:

$$\frac{\sqrt{n}\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)}{\sqrt{\underline{a}^{\mathrm{T}}\Sigma\underline{a}}} \sim N_1(0,1) \qquad \text{I}$$

Σ is not known, but we know $(n-1)S \sim Wish(\Sigma, \mathrm{n}-1)$

➔ $(n-1)\underline{a}^{\mathrm{T}}S\underline{a} \sim Wish\left(\underline{a}^{\mathrm{T}}\Sigma\underline{a}, \mathrm{n}-1\right) = (one\ dimensional\ wish\ is\ a\ chi\ square) = (\underline{a}^{\mathrm{T}}\Sigma\underline{a})\chi^2(\mathrm{p})$

Hence:

$$\frac{(n-1)\underline{a}^{\mathrm{T}}S\underline{a}}{\underline{a}^{\mathrm{T}}\Sigma\underline{a}} \sim \chi^2(p) \qquad \text{II}$$

I depend on $\overline{\underline{X}}$, II depends on S. $\overline{\underline{X}}\ and\ S\ are\ independent \rightarrow I\ and\ II\ are\ independent$

$$\frac{I}{\sqrt{\frac{II}{n-1}}} \sim t(n-1)$$

$$\frac{I}{\sqrt{\frac{II}{n-1}}} = \frac{\sqrt{n}\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)}{\sqrt{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}} \sim t(n-1) \text{ is pivotal}$$

➔ $$Pr\left[\frac{\sqrt{n}|\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}|}{\sqrt{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}} < t_{1-\frac{\alpha}{2}}(n-1)\right] = 1-\alpha$$

For better interpretation:

$$Pr\left[\underbrace{|\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}|}_{distance\ between\ what\ i\ want\ and\ the\ estimator} < \sqrt{\frac{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}{n}}t_{1-\frac{\alpha}{2}}(n-1)\right] = 1-\alpha$$

Two interpretations:

➔ Interval centred on $\underline{a}^T\underline{\mu}$, with $\underline{a}^T\overline{\underline{X}}$ having probability to stay in this interval $1-\alpha$

➔ Interval centred on $\underline{a}^T\overline{\underline{X}}$, with this interval having probability to cover $\underline{a}^T\underline{\mu}$ equal to $1-\alpha$

Confidence interval for linear combination of $\underline{\mu}$

The random interval covering $1-\alpha$ of the times the true linear combination of the mean.

$$CI_{1-\alpha}\left(\underline{a}^T\underline{\mu}\right) = \left[\underline{a}^T\overline{\underline{X}} \pm \sqrt{\frac{\underline{a}^T S\underline{a}}{n}}t_{1-\alpha}(n-1)\right]$$

(For the univariate case, you consider a with only a 1.)

Example for the difference $a = (0\ 0\ 0\ 1\ 0 \ldots 0 - 1\ 0 \ldots 0)$:

$$CI_{1-\alpha}\left(\underline{\mu_i} - \underline{\mu_j}\right) = \left[\overline{\underline{X_i}} - \overline{\underline{X_j}} \pm t_{1-\alpha}(n-1)\sqrt{\frac{S_{ii}+S_{jj}-2S_{ij}}{n}}\right]$$

More generally, if you take an $\underline{a}$ with 1 in position i, -1 in position j and another -1 in position k, you get:

$$CI_{1-\alpha}\left(\underline{\mu_i} - \underline{\mu_j}\right) = \left[\overline{\underline{X_i}} - \overline{\underline{X_j}} \pm t_{1-\alpha}(n-1)\sqrt{\frac{S_{ii}+S_{jj}+S_{kk}+2S_{jk}-2S_{ij}-2S_{ik}}{n}}\right]$$

Testing for linear combination of $\underline{\mu}$

Prove the linear combination is large (used for difference)

$$H_0: \underline{a}^T\underline{\mu} \leq \delta_0 \qquad vs \qquad H_1: \underline{a}^T\underline{\mu} > \delta_0$$

Test statistic:

$$\frac{\underline{a}^T\overline{\underline{X}} - \delta_0}{\sqrt{\underline{a}^T S\underline{a}}}\sqrt{n} = t_0$$

Reject if $t_0 > t_{1-\alpha}(n-1)$ $\quad (reject\ if\ the\ distance\ is\ too\ large)$

Prove the linear combination is different (used for comparison)

$$H_0: \underline{a}^T\underline{\mu} = \delta_0 \qquad vs \qquad H_1: \underline{a}^T\underline{\mu} \neq \delta_0$$

The test statistic is the same as before, but the rejection is different: Reject if $t_0 > t_{1-\frac{\alpha}{2}}(n-1)$

We proved that:

$$\forall \underline{a} \in R^p \qquad Pr\left[\underline{a}^T\underline{\mu} \in \left[\underline{a}^T\overline{\underline{X}} \pm \sqrt{\frac{\underline{a}^T S\underline{a}}{n}}t_{1-\alpha}(n-1)\right]\right] = 1-\alpha$$

This means that an $\underline{a}$ indicates one event. The probability of this event to occurs is $1-\alpha$.

But if I want to look at the probability of all this events? Do all this event simultaneously have probability to occur equal to $1-\alpha$? That is (this is false):

$$Pr\left[\underline{a}^T\underline{\mu} \in \left[\underline{a}^T\overline{\underline{X}} \pm \sqrt{\frac{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}{n}}t_{1-\alpha}(n-1)\right], \forall \underline{a} \in R^p\right] = 1-\alpha$$

We are saying: is it true that in any direction, I stay in that interval?
So that if I combine (for instance adding) two features, I stay in the interval with same confidence of staying in the difference between the two differences?
We need a larger interval to be sure at the same probability level, so $t_{1-\alpha}(n-1)$ will change.

Let's work on our pivotal squared, and we want to take its maximum, if is smaller than quantile:

$$\max_{\underline{a}\in R^p} n\frac{\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)^2}{\sqrt{\underline{a}^T S\underline{a}}}$$

For the maximization lemma:

$$\max_{\underline{a}\in R^p} n\frac{\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)^2}{\sqrt{\underline{a}^T S\underline{a}}} = n\left(\overline{\underline{X}}-\underline{\mu}\right)^T S^{-1}\left(\overline{\underline{X}}-\underline{\mu}\right)$$

And with the assumption of Gaussianity (for the hotelling's theorem) we know the distribution of this:

$$n\left(\overline{\underline{X}}-\underline{\mu}\right)^T S^{-1}\left(\overline{\underline{X}}-\underline{\mu}\right) \sim \frac{(n-1)p}{n-p}F(p,n-p)$$

Going to our problem of finding the simultaneous confidence interval, we need to find how much the interval is large, so **find *c*** that:

$$Pr\left[\underline{a}^T\underline{\mu} \in \left[\underline{a}^T\overline{\underline{X}} \pm \boldsymbol{c}\sqrt{\frac{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}{n}}\right], \forall \underline{a} \in R^p\right] = 1-\alpha$$

That is, squaring and putting on left side:

$Pr\left[n\frac{\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)^2}{\underline{a}^T S\underline{a}} \le c^2, \forall \underline{a} \in R^p\right] = 1-\alpha$ (here I'm saying: "everyone must be taller than c^2")

$$= Pr\left[\begin{array}{c}\underbrace{\max_{\underline{a}\in R^p} n\frac{\left(\underline{a}^T\overline{\underline{X}} - \underline{a}^T\underline{\mu}\right)^2}{\underline{a}^T S\underline{a}}} \\ \left(\overline{\underline{X}}-\underline{\mu}\right)^T S^{-1}\left(\overline{\underline{X}}-\underline{\mu}\right) for\ maximization\ lemma \\ \sim\frac{(n-1)p}{n-p}F(p,n-p)\ (the\ pivotal)\end{array} \le c^2\right] = 1-\alpha$$

(here I'm saying: "the tallest must be higher than c^2")

Hence:

$$Pr\left[\underline{a}^T\underline{\mu} \in \left[\underline{a}^T\overline{\underline{X}} \pm \sqrt{\frac{(n-1)p}{n-p}F_{1-\alpha}(p,n-p)}\sqrt{\frac{\underline{a}^{\mathrm{T}}\mathrm{S}\underline{a}}{n}}\right], \forall \underline{a} \in R^p\right] = 1-\alpha$$

This is **simultaneous confidence interval.**

All possible $\underline{a}$ and all simultaneously.

$$SimCI_{1-\alpha}\left(\underline{a}^T\underline{\mu}\right) = \underline{a}^T\overline{\underline{X}} \pm \sqrt{\frac{(n-1)p}{n-p}F_{1-\alpha}(p,n-p)}\sqrt{\frac{\underline{a}^T\underline{\underline{S}}\underline{a}}{n}}$$

Usually this is a larger interval respect to the one at a time interval.
Maybe too large, since you look for all of them simultaneously. We can relax this and look for a finite number of simultaneous events occurring at the same time, instead of looking at all possible directions.

Bonferroni

Looks for finite number of linear combinations.
For limited *(k)* events one at the time, we have with confidence β:

$$\underline{a}_1, \dots, \underline{a}_k \qquad CI_{1-\beta}\left(\underline{a}^T\underline{\mu}\right) = [\underline{a}^t\overline{\underline{X}} \pm t_{1-\frac{\beta}{2}}(n-1)\sqrt{\frac{\underline{a}^T\underline{\underline{S}}\underline{a}}{n}}]$$

And simultaneously I want:

$$Pr[\cap_i^k[\underline{a_i}^T\underline{\mu} \in CI_{1-\beta}\left(\underline{a_i}^T\overline{\underline{\mu}}\right)]] = 1-\alpha$$

$$opposite\ event = 1 - Pr[\cup_i^k[\underline{a_i}^T\underline{\mu} \notin CI_{1-\beta}\left(\underline{a_i}^T\overline{\underline{\mu}}\right)]] = 1 - \sum_i^k Pr\left[\underline{a_i}^T\underline{\mu} \notin CI_{1-\beta}\left(\underline{a_i}^T\overline{\underline{\mu}}\right)\right] = 1 - k\,\beta$$

Therefore, if you need k simultaneous confidence intervals, just take the univariate confidence interval for the linear combination of the mean, but with:

$$\beta = \frac{\alpha}{k}$$

$$BCI_{1-\alpha}(\underline{a}^T\underline{\mu})\ [\underline{a}^T\overline{\underline{X}} \pm t_{1-\frac{\alpha}{2k}}(n-1)\sqrt{\frac{\underline{a}^T\underline{\underline{S}}\underline{a}}{n}}]$$

Note: if k is very big you get a larger interval that the F interval. So, use it only if you have k small.

Bonferroni strategy for testing

Test against some predefined directions you're interested in.

$$H_0: \begin{cases} \underline{a_1}^T\underline{\mu} = \delta_1 \\ \dots \\ \underline{a_k}^T\underline{\mu} = \delta_k \end{cases} \qquad vs \qquad at\ least\ one\ of\ the\ previous\ row\ is\ false$$

Reject H_0 at level α if for at least one i:

$$\frac{|\underline{a_i}^T\overline{\underline{X}} - \delta_i|\sqrt{n}}{\sqrt{\underline{a_i}^T\underline{\underline{S}}\underline{a_i}}} > t_{1-\frac{\alpha}{2k}}(n-1)$$

Let's see the probability to reject when is actually true:

$$P[reject\ H_0|H_0\ is\ true] = P\left[\bigcup_i^k \frac{|\underline{a}_i{}^T\overline{\underline{X}} - \delta_i|\sqrt{n}}{\sqrt{\underline{a}_i{}^T S \underline{a}_i}} > t_{1-\frac{\alpha}{2k}}(n-1) \middle| H_0\ is\ true\right]$$

$$\leq \sum_i^k \underbrace{\Pr\left[\frac{|\underline{a}_i{}^T\overline{\underline{X}} - \delta_i|\sqrt{n}}{\sqrt{\underline{a}_i{}^T S \underline{a}_i}} > t_{1-\frac{\alpha}{2k}}(n-1)\right]}_{\frac{\alpha}{k}} = \alpha$$

➔ **If k is large, $t_{1-\frac{\alpha}{2k}}(n-1)\ gets\ big \rightarrow$ you'll never reject H_0**

- Is very conservative, hard to pass this test. So you'll never prove that your drug has effect.

14-3

Large scale hypothesis testing and false discovery rate (FDR)

k hypothesis to be tested simultaneously.

$$H_0\colon\ H_{01}\ \&\ H_{02}\ \&\ \ldots.\ \&\ K_{0k} \qquad vs \qquad H_1\colon H_{11}\ or\ H_{12}\ or\ \ldots.\ or\ K_{1k}$$

For each $i = 1, \ldots, k$ you may have a t statistic, with some p-value.
$Let\ p_i\ \ be\ the\ pvalue\ of\ H_{0i} vs\ H_{1i}$.

The Bonferroni strategy would be: take your probability (α) to reject at least one when is actually true, and reject $H_{0i}\ if\ p_i < \frac{\alpha}{k}$.

So as shown in the last example, you get that you must have a value greater than $t_{1-\frac{\alpha}{2k}}(n-1). If\ k \rightarrow \infty$ to reject a test is very hard. And **this is because you're running others test simultaneously:** is treating every test at the same importance. You reject a test because of the other…this can be too conservative.

$$P[make\ at\ least\ one\ mistake] = P[reject\ at\ least\ one\ H_0|\ all\ of\ H_{0i}\ are\ true] = \alpha\ (see\ before)$$

I can say only that I made at least one or more mistake with that probability.
I need something more.

False Discovery Rate (FDR)

$$H_0\colon\ H_{01}\ \&\ H_{02}\&\ \ldots.\&\ K_{0k} \qquad vs \qquad H_1\colon H_{11}\ or\ H_{12}\ or\ \ldots.or\ K_{1k}$$

$$Let\ D\ be\ any\ strategy(i.e.bonferroni)\ to\ run\ H_0 vs\ H_1$$

As column the Decision taken by the strategy.
As row the ground truth.

	Don't reject H_{0i}	Reject H_{0i}
H_0	U	V
H_1	T	S

V: false discovery (the discovery is the rejection, so I made a discovery, but is false)
S: true discovery
T: missed discovery. Number of discoveries I could have done, but I didn't since I've not rejected

R: the total rejected (= V+S)
K: the one not rejected (= U+T)
k_0: the true hypothesis

The only thing I can observe is K and R: indeed, I don't know the true and the false, but only what I reject or accept.
Say H_{0i} is true iff $i \in I_0\ \ so\ |I_0| = k_0$

We want to see what happens with Bonferroni, so $D = Bonferroni$

$$P[make\ at\ least\ one\ false\ discovery] = \cdots = as\ before = \sum_{j \in I_0} \frac{\alpha}{k} = k_0 \frac{\alpha}{k} \leq\ \alpha$$

Bonferroni is controlling the probability of at least one false discovery.
This is called Family Wise error rate.
Bonferroni control such that $FWER \leq\ \alpha$.

We want to control **False Discovery Rate**, that is $\frac{V}{R}$.

Definition of FDR:
Let:

$$Q = \begin{cases} 0 & if\ R = 0 \\ \dfrac{V}{R} & if\ R > 0 \end{cases}$$

$$FDR\ =\ E[Q]$$

Note

- ➔ **If all hypotheses are true**: our experiment should not produce any discovery.
  We should have $k_0 = k. and\ S = 0$. Therefore $V = R$.

$$Q = \begin{cases} 0 & if\ R = 0 \\ 1 & if\ R > 0 \end{cases}$$

Q is a Bernoulli variable, so

$$FDR = E[Q] = P[V > 0] = P[V \geq 1] = FWER$$

➔ If there are no discovery (never rejected), controlling FDR is the same as controlling FWER

➔ If there is something to discover $k_0 < k$.
Call and indicator variable for $V > 0$: $I = 0\ if\ V = 0 \quad and\ I = 1\ if\ V > 0$

$$Q \leq I$$

$$FDR = E[Q] \leq E[I] = P[V > 0] = P[V \geq 1] = FWER$$

$$FDR \leq FWER$$

Easier to control FDR than the FWER! Controlling FDR is less restrictive than controlling FWER.
So, if you have a strategy that control FWER at a level 50%, then for sure it will control the FDR at a level at least 50%.

I'm not anymore controlling the probability of sending an innocent to jail (rejecting H0 when is true), as Bonferroni was doing.
I'm controlling the ratio of innocent that are in jail: not controlling the individual discovery, but the average (E[Q]). Not controlling the Type 1 error.

A strategy to control FDR: B&H

Let p_i be the p-value for $H_{0i}\ vs\ H_{1i}$.
Order the k p-values growing: $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(k)}$ (the index is changed, not anymore corresponding to the hypothesis indexing).

Given an α.
Take $m = \max\{i \in \{1, \ldots, k\} : p_{(i)} \leq \frac{i\alpha}{k}\}$: $\quad \alpha\ and\ k\ are\ fixed, so\ is\ a\ line\ in\ function\ of\ i$

$$Reject\ H_{0i}\ \ if\ \ i < m$$

Theorem B&H:
If $p_1, \ldots, p_k\ are\ independent\ (pvalues\ are\ function\ of\ the\ data: are\ the\ probability\ to\ stay\ on\ the\ right)$, the strategy D_α that reject $H_{0(i)}\ if\ i < m$, controls FDR a level α

But what if $p_1, \dots, p_k$ *are not independent*? Like a testing impact another test?

Theorem B&Y:

- ➔ If $p_1, \dots, p_k$ *are positively correlated* (if a p-value increase also another one increase)
  $$\rightarrow use\ B\&H\ theorem$$
- ➔ If $p_1, \dots, p_k$ *are negatively correlated*
  - Consider this strategy:

$$Reject\ H_{0i}\ \ if\ \ i < m^* = \max\{j \in \{1, \dots, k\}: p_{(j)} < \frac{j}{C_{(k)}k}\alpha\}$$

$$where C_{(k)} = \sum_{j}^{k} \frac{1}{j}$$

And see that $C_{(k)}$ diverges, so for high k, the p-value must be very small.

There can be scenario where Bonferroni would reject everything, and B&H would accept some points. But some points may have high p-values, even if they reject H0 (with B&H strategy).

- ➔ A stronger version of B&H: **Efron's strategy**

Efron's strategy

$$Reject\ H_{0i}\ \ if\ p_{(i)} \leq i\frac{\alpha}{k}$$

Comparing means of a Multivariate Gaussian

Paired Data

We have n statistical units observed twice.
For a single unit i, I observe:

$$\underline{x}_{1i} \in R^p \quad and \quad \underline{x}_{2i} \in R^p$$

Where the first is, say p = 3, (weight salary and height for a person at 1-st).
The second is, say p = 3, (weight salary and height for a person at 2-nd time).

Features must be independent for the same instance (NOT among, there must be a dependence between weight at start time and weight at second time).
I assume instead $(\underline{x}_{11}, \underline{x}_{21}), \dots., (\underline{x}_{1n}, \underline{x}_{2n})\ \ iid \sim N_{2p}((\mu_1, \mu_2), \Sigma)$ with n observations.

We want to see if the diet had an effect on those features.
Call the difference:

$$D_i = \underline{x}_{1i} - \underline{x}_{2i}$$

The estimators:

$$\overline{\underline{D}} = \frac{1}{n}\sum_{i}^{n} D_i$$

$$S_D = \frac{1}{n-1}\sum_{i}^{n}(D_i - \overline{\underline{D}})(D_i - \overline{\underline{D}})^T$$

We want to test $H_0: \underline{\mu}_1 - \underline{\mu}_2 = \underline{\delta}$

Therefore:

$$CR_{1-\alpha}(\mu_1 - \mu_2) = CR_{1-\alpha}(\underline{\delta}) = \{\underline{\delta} \in R^p : n\left(\overline{\underline{D}} - \underline{\delta}\right)^T S_D^{-1}\left(\overline{\underline{D}} - \underline{\delta}\right) \leq \frac{(n-1)p}{n-p} F_{1-\alpha}(p, n-p)\}$$

So we reject H_0 if $n\left(\overline{\underline{D}} - \underline{\delta}\right)^T S_D^{-1}\left(\overline{\underline{D}} - \underline{\delta}\right) > \frac{(n-1)p}{n-p} F_{1-\alpha}(p, n-p)$

Ok we rejected H0, but what is the component that allowed me to reject? (Was it for the weight, for the height or the salary, for the sum of them?).

Simultaneous:

$$SimCI_{1-\alpha}(\underline{a}^T\underline{\delta}) = \{\underline{a}^T\overline{\underline{D}} \pm \sqrt{\frac{(n-1)p}{n-p} F_{1-\alpha}(p, n-p)} \sqrt{\frac{\underline{a}^T S \underline{a}}{n}}\}$$

It can happen that the mean of the vector is different, but the single component (weight) is not changed in the treatment. I proved that for some a of linear combination of the mean, there is change, not for each linear combination possible.

18-3

Repeated measurements, univariate case

$x_1, \dots, x_n$ indicates same quantity, observed in different times.

Assume $x_1, \dots, x_n \; iid \sim N_q\left(\underline{\mu}, \Sigma\right)$, indicate q measurements repeated on the same unit $i = 1, \dots, n$.

So $\underline{x_i} = \left(x_{i1}, \dots, x_{iq}\right) \in R^q$.

The mean of the same quantity on the repeated q measurements:

$$\underline{\mu} = \left(\mu_1, \dots, \mu_q\right)$$

We want to see if there was a change in this means, so:

$$H_0: \mu_1 = \mu_2 = \cdots = \mu_q \quad vs \quad H_1: \exists i, j: \mu_i \neq \mu_j$$

H_0 is equivalent to say that the $\underline{\mu}$ vector lives in the linear space generated by one ($L(\underline{1})$): linear space in R^q with all components equal.

$$H_0: \underline{\mu} \in L(\underline{1}) \quad vs \quad H_1: \underline{\mu} \notin L(\underline{1})$$

Translate the above in algebraic words:

$$Let\ C\ be\ a\ R^{(q-1)x\,q}\ contranst\ matrix, built\ as\ C = \begin{matrix} \underline{c}_1 \\ \vdots \\ \underline{c}_{q-1} \end{matrix}, \underline{c_i} \in R^q, with\ \underline{c}_1 \dots \underline{c}_{q-1}\ linear\ independent$$

And it becomes that H_0 tests whether $q-1$ combinations are linearly dependent.

$$H_0: C\underline{\mu} = \underline{0} \quad vs \quad H_1: C\underline{\mu} \neq 0$$

For the property of invariance of MLE: $C\overline{\underline{X}} \sim N_{q-1}(C\underline{\mu}, \frac{1}{n} C\Sigma C^{\mathrm{T}})$

And for Σ: $(n-1)CSC^T \sim Wish(C\Sigma C^{\mathrm{T}}, n-1)$ $independent\ from\ C\overline{\underline{X}}$

For hotelling:

$$\underbrace{n\left(C\overline{\underline{X}}-C\underline{\mu}\right)^T(CSC^T)^{-1}\left(C\overline{\underline{X}}-C\underline{\mu}\right)}_{pivotal} \sim \frac{(n-1)(q-1)}{n-q+1}F(q-1,n-q+1)$$

This is looking, as usual, how far is the estimate ($C\overline{\underline{X}}$) from the real value ($C\underline{\mu}$).

Under the hypothesis of H_0 we have our pivotal:

$T_0^2 = n\left(C\overline{\underline{X}}\right)^T(CSC^T)^{-1}\left(C\overline{\underline{X}}\right)$.

Reject at level α if:

$T_0^2 > \frac{(n-1)(q-1)}{n-q+1}F_{1-\alpha}(q-1,n-q+1)$.

But is the choice of the contrast matrix influencing the testing? See if for a different contrast matrix the test change:

$$Let\ C\ and\ \tilde{C}\ be\ 2\ contrast\ matrix, with\ C = B\tilde{C}.$$

$$T_0^2 = n\left(C\overline{\underline{X}}\right)^T(CSC^T)^{-1}\left(C\overline{\underline{X}}\right) = n\left(B\tilde{C}\overline{\underline{X}}\right)^T\left(B\tilde{C}SB\tilde{C}\right)^{-1}\left(B\tilde{C}\overline{\underline{X}}\right) = n\left(\tilde{C}\overline{\underline{X}}\right)^T B^T B^{T^{-1}}\left(\tilde{C}S\tilde{C}^T\right)^{-1}B^{-1}B\left(\tilde{C}\overline{\underline{X}}\right)$$
$$= n\left(\tilde{C}\overline{\underline{X}}\right)^T\left(\tilde{C}S\tilde{C}^T\right)^{-1}\left(\tilde{C}\overline{\underline{X}}\right)$$

Therefore is invariant from the representation.
If we take two matrix that span the same space, the statistic is the same.

There could be some contrast matrix more interesting than others. See notes for that.

Repeated measurements, multivariate case

Repeat not only a thing at each time, but two things. Say height and weight.
For a single unit i:

$$\begin{pmatrix} x_{i1}(height) \\ x_{i1}(weight) \end{pmatrix}, \begin{pmatrix} x_{i2}(height) \\ x_{i2}(weight) \end{pmatrix}, \dots, \begin{pmatrix} x_{iq}(height) \\ x_{iq}(weight) \end{pmatrix}$$

Units are independent, but features (generally) are not independent, also in a single unit.

$$H_0: \begin{cases} \mu_1(height) = \mu_2(height) = \cdots = \mu_q(height) \\ \mu_1(weight) = \mu_2(weight) = \cdots = \mu_q(weight) \end{cases}$$

The contrast matrix is a block matrix, with weight in a block and height in the other.
Change of representation:

$X_i = \left(X_{i1}(height), X_{i2}(height), \dots, X_{iq}(height), X_{i1}(weight), \dots, X_{iq}(weight)\right) \in R^{2q}$

MANOVA: Multivariate Analysis of Variance

Having g samples from a gaussian distributions, with different sizes, and independent:

$\underline{x}_{11}, \dots, \underline{x}_{1n_1}$ $iid \sim N_p\left(\underline{\mu}_1, \Sigma\right)$ measurement of p features in a sample, treated with a treatment 1

$\underline{x}_{21}, \dots, \underline{x}_{2n_2}\ iid \sim N_p\left(\underline{\mu}_2, \Sigma\right)$ measurement of p features in another sample, treated with a treatment 2

⋮

$\underline{x}_{g1}, \dots, \underline{x}_{gn_g}\ iid \sim N_p\left(\underline{\mu}_g, \Sigma\right)$ measurement of p features in another sample, treated with a treatment g

- There is independence because there are g different populations.
- **Units are independent too:** units in same group may come from Italy, Germany,...

The only thing that must be equal is the variance covariance matrix.

Why is called analysis of variance if we analyse the means? We want to analyse means of the populations to compare the variability:

- **The variability inside groups must be the same, the variability AMONG (between) groups is what we want to analyse.**
  We want to see the variability across the means of the groups.
  See the note MANOVA on tablet.

Goal: inference on $\underline{\mu}_1, \dots, \underline{\mu}_g$

Case $p \geq 1, g = 2$

We estimate the means of the groups:

$$\overline{\underline{X}}_1 = \frac{1}{n_1}\sum_{i=1}^{n_1} \underline{x}_{i1} \sim N_p\left(\underline{\mu}_1, \frac{1}{n}\Sigma\right)$$

$$\overline{\underline{X}}_2 = \frac{1}{n_2}\sum_{i=1}^{n_2} \underline{x}_{i2} \sim N_p(\underline{\mu}_2, \frac{1}{n}\Sigma)$$

And we want to see: $\overline{\underline{X}}_1 - \overline{\underline{X}}_2 \sim N_p\left(\underline{\mu}_1 - \underline{\mu}_2, \frac{1}{n_1}\Sigma + \frac{1}{n_2}\Sigma\right)$, that is unbiased for $\underline{\mu}_1 - \underline{\mu}_2$.

$$\left(\frac{1}{n_1} + \frac{1}{n_2}\right)^{-1}\left[\left(\overline{\underline{X}}_1 - \overline{\underline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right] \sim N_p(\underline{0}, \Sigma)$$

The sample covariances are:

$$S_1 = \frac{1}{n_1}\sum_j \left(x_{1j} - \overline{X}_1\right)\left(x_{1j} - \overline{X}_1\right)^T \sim \frac{1}{n_1 - 1} Wish(\Sigma, \mathrm{n}_1 - 1)$$

$$S_2 = \frac{1}{n_2}\sum_j \left(x_{2j} - \overline{X}_2\right)\left(x_{2j} - \overline{X}_2\right)^T \sim \frac{1}{n_2 - 1} Wish(\Sigma, \mathrm{n}_2 - 1)$$

$$S_1\ and\ S_2\ independent.$$

This is why we need same covariance of the groups.
They are both unbiased.

The weighted average of the sample covariance matrix is called S_{pooled} and is:

$$S_{pooled} = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{n_1 + n_2 - 2}$$

Therefore:

$$(n_1 + n_2 - 2)S_{pooled} \sim Wish(\Sigma, \mathrm{n}_1 + \mathrm{n}_2 - 2)$$

For hotelling's theorem, we get the pivotal:

$$\left(\frac{1}{n_1}+\frac{1}{n_2}\right)^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right]^T S_{pooled}^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right] \sim \frac{(n_1 + n_2 - 2)p}{n_1 + n_2 - 1 - p} F(p, n - p + 1)$$

That we use to check if difference of means is different:

$$CR_{1-\alpha}\left(\underline{\mu}_1 - \underline{\mu}_2\right) = \{\underline{\eta} = \underline{\mu}_1 - \underline{\mu}_2 \in R^q : \left(\frac{1}{n_1}+\frac{1}{n_2}\right)^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \underline{\eta}\right]^T S_{pooled}^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \underline{\eta}\right] \leq \frac{(n_1 + n_2 - 2)p}{n_1 + n_2 - 1 - p} F_{1-\alpha}(p, n - p + 1)\}$$

Or equivalent with hypothesis testing (H_0: $\underline{\mu}_1 - \underline{\mu}_2 = \delta_0$) reject at level α if:

$$\left(\frac{1}{n_1}+\frac{1}{n_2}\right)^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \delta_0\right]^T S_{pooled}^{-1}\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \delta_0\right] \geq \frac{(n_1 + n_2 - 2)p}{n_1 + n_2 - 1 - p} F_{1-\alpha}(p, n - p + 1)$$

And we use it to test if the means of the samples with different treatment is close or not.
We reject if is very unlikely that they're close.

Once we reject H_0, that means that the treatment had an effect, and so the means are far, we want to see "how far" they are. We build Simultaneous confidence intervals (and Bonferroni, for a fixed linear combination: the $\underline{\mu}_1 - \underline{\mu}_2$ combination).

But, we assumed $\Sigma_1 = \Sigma_2$. How can we say it?
H_0: $\Sigma_1 = \Sigma_2$.
This is a case of hypothesis testing where we don't want to reject.
Is not an easy test: can look at matrix of colours.

Mathematical ways:

1- Extension of Lovene's test
2- Take a distance between covariances: they are points on the Riemannian manifold of SPD matrix. Is not a vector space. Those distances can be calculated by approximations: approximate the sphere of the Riemannian manifold with plane. This plane is the vectorial space, where there are distance.

If the hypothesis is rejected, is still an open problem to solve (Behrens-Fisher problem).
We only know the solution of a specific case: when $n_1 \to \infty, n_2 \to \infty$.

When $n_1 \to \infty, n_2 \to \infty$ and the populations have different covariance

CLT:

$$\underline{\overline{X}}_1 \sim N_p\left(\underline{\mu}_1, \frac{1}{n_1}\Sigma_1\right)$$
$$\underline{\overline{X}}_2 \sim N_p\left(\underline{\mu}_2, \frac{1}{n_2}\Sigma_2\right)$$

$$\underline{\overline{X}}_1 - \underline{\overline{X}}_2 \sim N_p\left(\underline{\mu}_1 - \underline{\mu}_2, \frac{1}{n_2}\Sigma_2 + \frac{1}{n_1}\Sigma_1\right)$$

$$\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right]^T \left(\frac{\Sigma_1}{n_1} + \frac{\Sigma_2}{n_2}\right)^{-1} \left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right] \sim \chi^2(\mathrm{p})$$

This isn't pivotal since the Σ-s matrices aren't known, but for the LLN, we have:

$$\left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right]^T \left(\frac{S_1}{n_1} + \frac{S_2}{n_2}\right)^{-1} \left[\left(\underline{\overline{X}}_1 - \underline{\overline{X}}_2\right) - \left(\underline{\mu}_1 - \underline{\mu}_2\right)\right] \sim \chi^2(\mathrm{p})$$

That is pivotal.

Lexicon:

- 2 way: 2 treatments
- g levels (g populations with different levels of drug)
- b levels for the second treatment(if is a 2 way)

19-3

One way ANOVA

Case when there is only a treatment (one way) and lots of populations $\boldsymbol{(g \geq 1)}$. For each population, there are different levels of treatment (different doses of drug).
Is univariate, so $\boldsymbol{p = 1}$. p is not about the levels of treatment! Is like saying look for glucose level affected by a drug.

For first population treated with drug at level 1:

$$x_{11}, \dots, x_{1n_1} \; iid \sim N_1(\mu_1, \sigma^2)$$

For second population treated with drug at level 2:

$$x_{21}, \dots, x_{2n_2} \; iid \sim N_1(\mu_2, \sigma^2)$$

And so on… for:

$$x_{g1}, \dots, x_{gn_g} \; iid \sim N_1(\mu_g, \sigma^2)$$

All those populations are **independent** and **with same variance.** Also, the components of the samples are independent (since iid). For instance, you measure put into a group person of different countries.

Notation:

- i: index of the treatment (from 1 to g)
- j: index of the population (from 1 to n_i)

We parametrize the problem in a different way, where the mean is represented with parameter:

- μ: the **general (overall) mean** that doesn't change from treatment to treatment
- τ_i: the **effect on the mean produced by the treatment**. Depends on the level of the treatment i. Tells us how fare are we from the overall mean because of the treatment.
- With $\varepsilon_{ij} \; iid \sim N\,(0, \sigma^2)$ accounting for variability. Residual part. Doesn't impact the mean.

$$X_{ij} = \mu + \tau_i + \varepsilon_{ij}$$

We have to estimate those parameters.

The mean can be rewritten as $\mu_i = \mu + \tau_i$. In this formulation we have $g + 1$ parameters $(\mu, \tau_1, .., \tau_g)$.
But in the original one we had g parameters $(\mu_1, \dots, \mu_g)$.
Therefore, the problem is overparametrized, introduce a constraint on the parameters, see later how.

Estimator for the overall mean μ

This characterizes the overall population, even among treatments.
To do this, have to take every point of the dataset:

$\overline{X} = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n_g} x_{ij}$ estimator for μ Where $n = n_1 + n_2 + \cdots + n_g$

Is it unbiased?

$$E[\overline{X}] = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n_g} E[x_{ij}] = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n_g}(\mu + \tau_i) = \frac{1}{n}\sum_{i=1}^{g} n_i\,\mu + \frac{1}{n}\sum_{i=1}^{g} n_i\,\tau_i = \mu + \frac{1}{n}\sum_{i=1}^{g} n_i\,\tau_i$$

To make it unbiased, $\frac{1}{n}\sum_{i=1}^{g} n_i\,\tau_i$ must be null. Therefore, impose a constraint on $\tau_1, \dots, \tau_g$:

$$\sum n_i \tau_i = 0$$

So if you know all but not one τ_i, you can get it from this constraint.
The effect on the mean are linearly dependent.

Estimator for the treatment effect τ

We want to see how far is the mean of the group from the overall mean (estimated as above) is:

- $\overline{X}_i - \overline{X}$

We see that is unbiased:

$$E[\overline{X}_i - \overline{X}] = \underbrace{E[\overline{X}_i]}_{\mu + \tau_i} - \underbrace{E[\overline{X}]}_{=\mu} = \tau_i \quad \text{if constraint is satisfied}$$

If we have the same number of observations in each group (balanced design, $n_1 = n_2 = \cdots = n_g$), then

$$\sum n_i \tau_i = \sum \tau_i = 0$$

In this way the estimators are unbiased.
If you use this constraint when the design is not balanced, the estimators become biased.

If the treatment produces an effect, that effect should be on the mean. So we want to prove there was an effect:

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

That is equivalent to:

$$H_0: \tau_1 = \tau_2 = \cdots = \tau_g \quad vs \quad H_1: \exists\, \tau_j \neq 0$$

Decomposition of variance.

Writing a vector, built like this:

$$\underline{X} = (x_{11}, \dots, x_{1n_1}, x_{21}, \dots, x_{2n_2}, \dots., x_{g1}, \dots, x_{gn_g}) \in R^n$$

We use this utility vectors, with the first n_i components equal to 1, and the rest equal to 0:

$$\underline{u}_1 = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \underline{u}_2 = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \end{pmatrix} \textit{ and so on}$$

$\underline{u}_1$ is the vector indicating the linear space where the first n_1 compoents are the same: where there is no variability for the group 1.
And so on for each group.

$$\underline{u}_i \in R^n \quad \forall i$$

Those vectors $\underline{u}_1, \dots, \underline{u}_g$ are linearly independent, orthogonal and they span a linear space with dimension g, where $\underline{1}$ belongs in. Because $\underline{1} = \sum_{i=1}^{g} \underline{u}_i$.

I can project our vector $\underline{X}$ on the linear space generated by those utility vector. Since those vectors are orthogonal, I can sum the projection on $\underline{u}_1$ plus the projection on $\underline{u}_2$ and so on, to obtain the sum of the projected vectors:

$$\pi_{\underline{X} \mid \underline{u}_1,\dots,\underline{u}_g} = \sum_{i=1}^{g} \frac{\underline{u}_i \underline{u}_i^T}{\underline{u}_i^T \underline{u}_i} \underline{X} = \sum_{i=1}^{g} \frac{1}{n_i} \sum_{j=1}^{n_i} \underline{X}_{ij} \underline{u}_i = \sum_{i=1}^{g} \overline{X}_i \underline{u}_i$$

This means to project the vector in a group-wise manner (I'm projecting over the space generated by the classes).
If we project our vector $\underline{X}$ on the $\underline{1}$ vector:

$$\pi_{\underline{X} \mid \underline{1}} = \frac{\underline{1}\,\underline{1}^T}{\underline{1}^T \underline{1}} \underline{X} = \overline{X}\,\underline{1}$$

Let's try to close the triangle: project the first projection on the second one (where there isn't variability):

$$\pi_{\sum_{i=1}^{g} \overline{X}_i \underline{u}_i \mid \underline{1}} = \overline{X} * \underline{1}$$

So we can close the triangle!

Note: all of these are orthogonal projections. This means that we can represent our $\underline{X}$ as **3 orthogonal vectors:**

$$\underline{X} = \overline{X} * \underline{1} + \sum_{i=1}^{g} (\overline{X}_i - \overline{X}) \underline{u}_i + \underline{X} - \sum_{i=1}^{g} \overline{X}_i \underline{u}_i$$

Degrees of freedom are:

$$n = 1 + g - 1 + n - g$$

All of these components are orthogonal vectors, so we can say that the norms:

$$\left|\left|\underline{X}\right|\right|^2 = \left|\left|\overline{X} * \underline{1}\right|\right|^2 + \left|\left|\sum_{i=1}^{g} (\overline{X}_i - \overline{X}) \underline{u}_i\right|\right|^2 + \left|\left|\underline{X} - \sum_{i=1}^{g} \overline{X}_i \underline{u}_i\right|\right|^2$$

And we get the **decomposition of variance formula 1** that is:

$$\underbrace{\sum_{i=1}^{g} \sum_{j=1}^{n_i} X_{ij}^2}_{SS_{total}} = \underbrace{n\overline{X}^2}_{SS_{mean}} + \underbrace{\sum_{i=1}^{g} (\overline{X}_i - \overline{X})^2 n_i}_{SS_{treatment}} + \underbrace{\sum_{i=1}^{g} \sum_{j=1}^{n_i} (X_{ij} - \overline{X}_i)^2}_{SS_{residual}}$$

The portion of the variance $SS_{treatment}$ is not null if you have some effect given by treatment.

- ➔ $SS_{treatment}$ is the variability introduced because there are multiple levels of treatment.
- ➔ $SS_{residual}$ is what is not captured by the linear combination of $\underline{u}_1, \dots, \underline{u}_g$ with $\underline{X}$.

But we have Pitagora also on the triangle in the picture, so we can write the **second version of the decomposition of variance**:

$$SS_{centered} = SS_{treatment} + SS_{residual}$$

- $SS_{centered}$ is the dispersion around the overall mean. Overall dispersion

- **The variability you have around the overall mean is due to the treatment: if there weren't variability, every point in each population would be exactly the mean. So the variability of the means of the populations around the overall mean is what show up due to the treatment. This is** $SS_{treatment}$.
- $SS_{residual}$ is the true variability inside groups. We've assumed it was equal for each group. Each group contributes to the residual variability. $SS_{residual}$ $estimate$ σ^2

Back to the problem, I want to test if there was an effect due to the treatment. I'd like to conclude that there is a variability on the means of the group, due to the treatment.
To assess that, the $\boldsymbol{SS_{treatment}}$ **must be large**. Large with **respect to the variability inside the group** (σ^2).

Note: if you have 2 groups, this reduces to t- test to compare two means!

IDEA: reject H_0 if $SS_{treatment}$ (the variability introduced by the treatment) is large with respect to $SS_{residual}$.
Geometrically this means:

- Reject if the difference (the thing that indicate the treatment effect) of the projection of $\underline{X}$ into the $\underline{1}$ vector (that indicate no variability in groups, since you project on the space considering equally all treatments, $\underline{1}$) with the projection of $\underline{X}$ into the space generated by $Span\{u_i\}$, is big. Big with respect to what? With respect to the projection of $\underline{X}$ into the space generated by $Span\{u_i\}$, that is the variability inside each group.

If is high enough to be not by chance.

So our test is: we want to see $\frac{SS_{treatment}}{SS_{residual}}$. If this is big we reject. But big in the distribution, seeing in which tail we are.
So we need to find the distribution of this ration when H_0 is true.

- $SS_{residual} = \sum_{i=1}^{g}\sum_{j=1}^{n_i}\left(X_{ij} - \overline{X}_i\right)^2 = \sum_{i=1}^{g}(n_i - 1)S_i^2 \sim \sigma^2\chi^2(\mathrm{n} - \mathrm{g})$
  - Because for a sample of size n, iid gaussian with same mean and variance:
    $S_i^2 = \frac{1}{n_i - 1}\sum_{j=1}^{n_i}\left(\underline{x}_j - \overline{x}_i\right)^2$ and we know that $(n-1)S^2 \sim \sigma^2\chi^2(\mathrm{n}-1)$.
    We have g groups, with same variance, and are independent. So we estimate each group's variance and we pool all the estimates. The pooled estimator for the variance is the weighted mean of the estimator in every group:
    $$S_{pooled} = \frac{1}{n-g}\sum(n_i - 1)S_i^2$$
    Summing g independent chi squares, each with degree: $n_i - 1$:
    $Repeat\ g\ (i.e. = 4) times: n_1 - 1 + n_2 - 1 + n_3 - 1 + n_4 - 1 = n - 4 = n - g$

If H0 is true: ($\mu_1 = \mu_2 = \cdots = \mu_g$) (the previous point is true no matter if H0 is true)
We need the hypothesis to be true, since we need distributions to be gaussians with those means.

- $SS_{centered} = \sum_{i=1}^{g}\sum_{j=1}^{n_i}\left(X_{ij} - \overline{X}\right)^2 = (n-1)S^2 \sim \sigma^2\chi^2(\mathrm{n}-1)$

$$SS_{treat} = ?$$

$$SS_{res} \text{ and } SS_{treat} \text{ are ortogonal vectors, we're in gaussian world} \rightarrow \text{they are independent}$$

So, $SS_{treat} \sim \sigma^2\chi^2(g-1)$, because $[n-1 = g-1+n-g]$

Therefore, (remember that they are independent)

$$\frac{\frac{SS_{treat}}{g-1}}{\frac{SS_{res}}{n-g}} \sim F(g-1, n-g)$$

So, reject H_0 *at level* α, saying that there was an effect on the mean because of the treatment if:

$$\frac{\frac{\sum_{i=1}^{g} n_i(\overline{x}_i - \overline{x})^2}{g-1}}{\frac{\sum_{i=1}^{g}\sum_{j=1}^{n_i}(\overline{x}_{ij} - \overline{x}_i)^2}{n-g}} > F_{1-\alpha}(g-1, n-g)$$

We reject if the treatment introduce variability.

MANOVA

It means $p \geq 1, g \geq 1$. So is multivariate.

$$x_{11}, \dots, x_{1n_1} \; iid \sim N_p\left(\underline{\mu}_1, \Sigma\right)$$
$$x_{21}, \dots, x_{2n_2} \; iid \sim N_p\left(\underline{\mu}_2, \Sigma\right)$$
$$\vdots$$
$$x_{g1}, \dots, x_{gn_g} \; iid \sim N_p\left(\underline{\mu}_g, \Sigma\right)$$

Reparametrizing:

$\underline{X}_{ij} = \underline{\mu} + \underline{\tau}_i + \underline{\varepsilon}_{ij}$, with $i = 1, \dots, g \qquad j = 1, \dots, n_i, \qquad \underline{\varepsilon}_{ij} \; iid \sim N_p(\underline{0}, \Sigma)$

$$\underline{\mu}, \underline{\tau}_i \in R^p$$

$$\sum_{i=1}^{g} n_i \underline{\tau}_i = \underline{0}$$

If we fix a component $k \in \{1, \dots, p\}$ is the same of ANOVA. Component wise, if $\Sigma = [\sigma_{ij}^2]$:

$\underline{X}_{ijk} = \underline{\mu}_k + \underline{\tau}_{ik} + \underline{\varepsilon}_{ijk} \qquad with\, \underline{\mu}_k, \underline{\tau}_{ik} \in R \quad \underline{\varepsilon}_{ijk} \; iid \sim N_1(0, \sigma_{ij}^2)$

So the multivariate model include multiple univariate models.
If you run it component wise, is equivalent to ANOVA. **But** you'd **miss** the **correlation** between components, not including Σ! You'd **use only the diagonal part of Σ**.
Of course, MANOVA is more expensive. So, if you have off-diagonal terms of the covariance matrix equal to zero, use ANOVA. Or you can perform PCA moving to a space with no correlation between components (eigenvectors are orthogonal) and then do separate ANOVAs on score on the PCA.

But if correlation is part of the problem, you should do MANOVA on original variables.

Remark:

- Total mean: $\underline{\overline{X}} = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n_i}\underline{x}_{ij}$
- Mean of group i: $\underline{\overline{X}}_i = \frac{1}{n_i}\sum_{j=1}^{n_i}\underline{x}_{ij}$

Decomposition of covariance

$$\underbrace{\sum_{i=1}^{g}\sum_{j=1}^{n_i}(\underline{x}_{ij}-\underline{\overline{x}})(\underline{x}_{ij}-\underline{\overline{x}})^T}_{Total\ covariance} = \underbrace{\sum_{i=1}^{g} n_i(\underline{x}_i-\underline{\overline{x}})(\underline{x}_i-\underline{\overline{x}})^T}_{Covariance\ due\ to\ the\ treament} + \underbrace{\sum_{i=1}^{g}\sum_{j=1}^{n_i}(\underline{x}_{ij}-\underline{\overline{x}}_i)(\underline{x}_{ij}-\underline{\overline{x}}_i)^T}_{Covariance\ due\ to\ residual}$$

$Covariance\ due\ to\ the\ treament$: $\boldsymbol{B}$ since is the between group covariance. **Note**: is counting the covariance n_i times for each group.
$Covariance\ due\ to\ the\ residual$: $\boldsymbol{W}$ since is the within group covariance. Compute the covariance for each unit, comparing with the mean, assuming variance/covariance is the same for every group, so only sum, without weighting it.
In fact, we can rewrite W (pooled estimate of the variance/covariance):

$$W = \sum_{i=1}^{g}(n_i-1)S_i$$

Proof of the decomposition of variance:

- $\underline{x}_{ij}-\underline{\overline{x}} = \left(\underline{\overline{x}}_i-\underline{\overline{x}}\right) + \left(\underline{\overline{x}}_{ij}-\underline{\overline{x}}_i\right)$ …. Do the algebra in the decomposition of variance formula

Our goal is to prove the effect of the treatment, so reject al level α:

$$H_0: \underline{\mu}_1 = \underline{\mu}_2 = \cdots = \underline{\mu}_g$$

That is equivalent to:

$$H_0: \underline{\tau}_1 = \underline{\tau}_2 = \cdots = \underline{\tau}_g \quad vs \quad H_1: \exists\, \underline{\tau}_j \neq \underline{0}$$

If we follow the same reasoning of ANOVA, we'd say: reject if the covariability due to treatment is big with respect to the covariability due to the residual. But those are matrixes! $W/B\ is\ large \rightarrow reject$.

There are multiple ways to assess this.

Wilks's proposal:
$\Lambda_{\mathrm{w}} = \frac{Det(W)}{Det(W+B)}$ and is like saying $\frac{1}{1+\frac{B}{W}}$.
We reject if $\boldsymbol{\Lambda_{\mathrm{w}}}$ is small.

Pillai's proposal:
$\Lambda_{\mathrm{p}} = tr(B(B+W)^{-1})$
Trace is the total variability.
We reject if $\boldsymbol{\Lambda_{\mathrm{p}}}$ is large.
Hotelling's proposal:
$\Lambda_{\mathrm{h}} = tr(BW^{-1})$
We reject if $\boldsymbol{\Lambda_h}$ is large.

All those statistics can be expressed in terms of eigenvalues of BW^{-1}.

$$\{\lambda_1, \dots., \lambda_s\}, where\ s = min(g-1, p)$$

Meaning the rank of BW^{-1} can't be less than $g-1$ since you're summarizing the entire data with at least g means ($\overline{X}_1, \dots., \overline{X}_g$), if *p* is very large.
If p is not large, then you have at least $g-1$ eigenvectors.
We don't know the distributions of this object Λ when H_0 is true. We only know it asymptotically.

21-3

MANOVA with g groups with each group having n_i observations.
Does membership in group i modify the mean?

Testing if treatment induce an effect:

$$H_0: \underline{\tau}_1 = \underline{\tau}_2 = \cdots = \underline{\tau}_g = \underline{0}\ (no\ treatment\ genereted\ a\ difference)\ vs \quad H_1: \exists\ \underline{\tau}_j \neq 0$$

If H_0 is true:

- **What is the distribution of** $\Lambda_w = \frac{Det(W)}{Det(W+B)}$, with the idea of reject H_0 if Λ_w is small? (reject if you're on the left tails of its distribution)
    - The problem: we don't know its distribution, apart from some cases:
        - $\boldsymbol{g = 2, 3\ and\ p \geq 1}$
        - $\boldsymbol{g \geq 1 \quad and\ p = 2}$

However, **we know the distribution of Λ if the sample size is large (and H_0 is true)**
Bartlett's asymptotic approximation

$$-\left(n - 1 - \frac{p+g}{2}\right) \log\ \Lambda_w \sim \chi^2(p(g-1))$$

Therefore, reject H_0 at level α if $-\left(n - 1 - \frac{p+g}{2}\right) \log\ \Lambda_w > \chi^2_{1-\alpha}(p(g-1))$

Since we want to reject if Λ_w is small, and so, with negative sign, the whole quantity is large.

Permutational test's idea: we don't know the distribution of the samples. If the test has no effect, you'd be able to permute the data across groups, and computing the t statistic for the mean. If you see a difference -> reject.

If you reject H_0, you want to see the effect generated by the treatment: confidence interval for the effect.
Confidence intervals for the differences:

$$\tau_{il} - \tau_{kl}$$

Where: i, k are the indexes of the groups. Meaning that we compare the group i and the group k.
And l is the component of the sample: $l = 1, \dots, p$.
So: is the different between group i and group k significant for the l feature (height or weight or salary or…)?

$$l = 1 \dots p \quad index\ of\ the\ features$$
$$k = 1 \dots g \quad index\ of\ the\ treatment\ level$$

We estimate those two as:

- $\tau_{il} \rightarrow (\overline{X}_{il} - \overline{X}_{l})$
- $\tau_{kl} \rightarrow (\overline{X}_{ik} - \overline{X}_{l})$

Therefore, for the difference $\tau_{il} - \tau_{kl}$:

$$(\overline{X}_{il} - \overline{X}_{l}) - (\overline{X}_{ik} - \overline{X}_{l}) = \overline{X}_{il} - \overline{X}_{ik}$$

The distribution of $\tau_{il} - \tau_{kl}$, since is a linear combination of gaussian variables:

$$\overline{X}_{il} - \overline{X}_{kl} \sim N_1((\tau_{il} - \tau_{kl}), \sigma_{ll}\left(\frac{1}{n_i} + \frac{1}{n_k}\right))$$

$$\sigma_{ll}\ belongs\ to\ the\ diagonal\ of\ \Sigma \rightarrow \text{we have to estimate } \Sigma$$

Noticing that $W = \sum_{i=1}^{g}(n_i - 1)S_i$, where S_i are estimator of covariance Σ in every group.

➔ Properly normalizing the DoFs:

$$S_{pooled} = \frac{1}{n-g}\sum_{i=1}^{g}(n_i - 1)S_i \quad estimating\ \Sigma$$

➔ Estimator for σ_{ll} is $\frac{W_{ll}}{n-g}$ where W_{ll} is the diagonal element of Σ.

I'm computing, given the two groups, the confidence intervals for p components.
The diffences possibles are $\frac{g(g-1)}{2}$, since you fix a choice (i) and then you have the other choice (j) that you can choose apart from the previously chosen (i). The ½ term is because differences are symmetric. So I have to divide Bonferroni by $\frac{pg(g-1)}{2}$

➔ **Bonferroni Simultaneous confidence intervals for the $\tau_{il} - \tau_{kl}$:**

$$SimCI(\tau_{il} - \tau_{kl}) = [\overline{X}_{il} - \overline{X}_{kl} \pm t_{\frac{\alpha}{2pg(g-1)}, \underbrace{(n-g)}_{\substack{g\ DoFs\ to\ estimate\ the\ means \\ n\ DoFs\ to\ estimate\ covariance}}}\sqrt{\frac{1}{n-g}W_{ll}\left(\frac{1}{n_i} + \frac{1}{n_k}\right)}]$$

Two-way ANOVA

Two treatments (called also factors), each one with different levels.
We want to find the combination of two treatments for the best effect on the mean.
Say:

- treatment 1 with g levels

- treatment 2 with b levels

There are $g * b$ combinations for the treatment's levels.
Assume a balanced experiments: $for\ each\ i,j$ (for each combination of levels of the two treatments) the same number of observations. (in principle, you'd have n_{ij} for each interception of the two treatment, but we say $n_{ij} = n\ \forall\ i,j$).

Remark: $X_{ijk}\ univariate$.

We can express:

$$X_{ijk} = \mu_{ij} + \varepsilon_{ijk}$$

- $i = 1, \dots, g$
- $\varepsilon_{ijk} \sim N(0, \sigma^2)\ univariate$

We need to see when the mean change due to both the level of treatment 1 and the level of treatment 2.
Fix the baseline μ, see the changes due to combination of the treatments at two different levels.
We have two possibilities:

- complete model
- additive model

Additive model, not considering interactions between treatments:
τ_i: effect of treatment 1 at level i
β_j: effect of treatment 2 at level j

$$X_{ijk} = \underbrace{\mu + \tau_i + \beta_j}_{\mu_{ij}} + \varepsilon_{ijk}$$

If you fix the effect of a treatment (say the one τ_1), then μ_{1j} depends only on the effect of the treatment 2 at different levels β_j. Plot μ_{ij} against β_j fixing $i = 1$:

$$\mu_{1j} = \mu + \tau_1 + \beta_j$$

Then for $i = 2$:

$$\mu_{2j} = \mu + \tau_2 + \beta_j$$

On the plot is only a translation: we discarded interactions.

➔ The plot shows how the mean varies when you fix the level for a treatment and let the other level of treatment change

The effect of fertilizer is the same no matter how much water you give.

Complete model, considering interactions between treatments:

$$X_{ijk} = \underbrace{\mu + \tau_i + \beta_j + \gamma_{ij}}_{\mu_{ij}} + \varepsilon_{ijk}$$

Now we're considering interactions. Therefore, the curve showing the effect on the means of the two treatments intercepts: one can go up and the other down. If you fix τ_i ($or\ \beta_j$), there is this term γ_{ij} still.

➔ The curves that describe how the mean varies when you fix the level for a treatment and let the other change, can intercept.
➔ **The effect of fertilizer is depending also on how much water you give.** It's more realistic, since if you don't give water, plants don't grow. But also, if you give it too much.

Price to pay for this model: you won't have enough data to estimate the variability.

Fix the treatment 1's level at 1 and see the effect of treatment 2.
$\overline{X}_{1*} = \frac{1}{nb}\sum_{j=1}^{b}\sum_{k=1}^{n} X_{1jk}$

Fix the treatment 2's level at 1 and see the effect of treatment 1.
$\overline{X}_{*1} = \frac{1}{ng}\sum_{j=1}^{g}\sum_{k=1}^{n} X_{i1k}$

The mean inside the cell:

$$\overline{X}_{ij} = \frac{1}{n}\sum_{k=1}^{n} X_{ijk}$$

Overall mean:

$$\overline{X} = \frac{1}{gbn}\sum_{i=1}^{g}\sum_{j=1}^{b}\sum_{k=1}^{n} X_{ijk}$$

Params	**Estimators**
μ	$\overline{X}$
τ_i	$\overline{X}_{i*} - \overline{X}$
β_j	$\overline{X}_{*j} - \overline{X}$
γ_{ij}	$(\overline{X}_{i*} - \overline{X})(\overline{X}_{*j} - \overline{X}) - \overline{X}$ $= \overline{X}_{ij} - \overline{X}_{i*} - \overline{X}_{*j} + \overline{X}$

Constraints

The number of parameters:

$$\mu_{ij} = \underbrace{\mu}_{1} + \underbrace{\tau_i}_{g} + \underbrace{\beta_j}_{b} + \underbrace{\gamma_{ij}}_{gb}$$

Since experiment is balanced:

$$0 = \underbrace{\sum \tau_i}_{1\ constraint} = \underbrace{\sum \beta_j}_{1\ constraint} = \underbrace{\sum_i \gamma_{ij} = \sum_j \gamma_{ij}}_{g+b-1\ constraints} \quad \forall i,j$$

Total constraints: $g + b + 1$ parameters constrained, the one we needed (see two row above).

Decomposition of variance

In treatment 1, we have a total of nb observations (considering all the combinations of the 2 treatments).
In treatment 2, we have a total of ng observations.
We have a total of n observations for each interaction (for each combination).

$$\underbrace{\sum_{i=1}^{g}\sum_{j=1}^{b}\sum_{k=1}^{n}\left(X_{ijk}-\overline{X}\right)^2}_{SS_{centred}}=\underbrace{\sum_{i=1}^{g}\left(\overline{X}_{i*}-\overline{X}\right)^2 nb}_{SS_{treatment_1}}+\underbrace{\sum_{j=1}^{b}\left(\overline{X}_{*j}-\overline{X}\right)^2 ng}_{SS_{treatment_2}}+$$
$$\underbrace{\sum_{i=1}^{g}\sum_{j=1}^{b}\left(\overline{X}_{ij}-\overline{X}_{i*}-\overline{X}_{*j}+\overline{X}\right)^2 n}_{SS_{interaction}}+\underbrace{\sum_{i=1}^{g}\sum_{j=1}^{b}\sum_{k=1}^{n}\left(X_{ijk}-\overline{X}_{ij}\right)^2}_{SS_{residual}}$$

So the variability is splitted in those terms.

The degrees of freedom are (remark that n is the number of observations per combination of treatments):

$$\underbrace{gbn-1}_{£}=\underbrace{(g-1)}_{Because\ of\ linear\ constraint}+\underbrace{(b-1)}_{same}+(g-1)(b-1)+gb(n-1)$$

£: total is gbn, **but one DoF taken to estimate the mean (the $-\overline{X}$ *term in* $SS_{centred}$): projected into linear space generated by $\underline{1}$.**

I've split the space orthogonal to the space generated by $\underline{1}$ (so $gbn-1$, the screen is gbn), into 4 orthogonal linear spaces where you can look at different effects (treatment 1, 2, interaction and residual).

If $n=1$: for every combination of treatments levels, there is only one observation.

- ➔ **Zero DoF for $SS_{residual}$: means that the model perfectly interpolates the data, with no variability. Overfitting.** You don't know the variability (is not that is zero, but that you don't know because can't be estimated)
  - You've estimated the effects of the treatments, but you don't know if they are relevant since you can't compare to $SS_{residual}$!
    - Example: for this combination I see a crop that is 100kg for m^2. For this other combination I see 120kg for m^2. Is there a difference between the two? I don't know, it depends on sigma. If it is 100kg, then no difference. If is 1kg, a lot.

Two possibilities to solve this:

- **Have at least $n=2$.** At least two observations per combination of treatment. (That translates into more money for the experiment)
- You can't explain simultaneously interaction and residual: the model will interpolate the data.

Bias variance trade-off:
You can either explain well the model ($SS_{treatment_1}+SS_{treatment_2}+SS_{interaction}$), or very well the variability.

Test if there is interaction between treatments

$$H_0: \gamma_{ij}=0$$

Reject when:

$$\frac{\frac{SS_{interaction}}{(g-1)(b-1)}}{\frac{SS_{residual}}{gb(n-1)}} > F_{1-\alpha}((g-1)(b-1), gb(n-1))$$

You hope to not reject in this case, unless you want to prove there is interaction.
Typically, you want no interaction to use additive model.

If you can't reject, it means interactions can be discarded.

➔ Use **additive model**: you can explain the variability without considering the interactions. So, **the term $SS_{interaction}$ goes into $SS_{residual}$**. Hence, you **can explain more variability**. You have more DoF to estimate the variance: $SS_{residual}$ **has now** $(g-1)(b-1)+gb(n-1)$. You can estimate variance even with a single observation.

$$F_0 = \frac{\frac{SS_{treatment}}{g-1}}{\frac{SS_{res}}{gbn-g-n+1}}$$

➔ Reject if $F_0 > F_{1-\alpha}(g-1, gbn-g-b+1)$

25-3

Each unit in a population (not in sample) is represented by a vector of feature and a label.
$(\underline{X}, L)$ with $\underline{X} \in R^p, \quad L = \{1, \dots, g\}$
L is representing the value of a possible subgroup of the population (male/female. Or male in milan, male in Paris. General)

Classification means learn a (decision) function δ that maps $\underline{X}$ into its label. So mapping the space of features into the space of labels.

$$\delta: R^p \to R, \quad hence\ \delta(\underline{X}) = l \quad with\ being\ the\ group\ observed\ for\ \underline{X}$$

Two ways:

- **Supervised** learning (discriminant analysis)
  Here you have the training data with the true label associated with the feature vector observed. $(\underline{X}, L)$
  You must associate the feature vector to the label. Not only on the training set, but for a new (unobserved) observation. (bias-variance trade-off)
- **Unsupervised** learning (cluster analysis)
  Here you have only features $\underline{X}$, and you must learn what are the labels L associated.
  Labels are hidden: there are labels, but you can't observe them. You believe there are labels associated with the observations, therefore, you have a prior assumption.
  The procedure is to cluster the units basing on some similarity of the observations. *Similar observations must belong to same group*. The hard part is define the similarity: on what label

are they similar? How much must they be *similar* in order to belong to the group?
Goals:

- Estimate labels $l_1, \dots, l_n$
- Estimate g

Then perform supervised classification with the estimated labels and group.

Supervised learning

Ingredients to build δ

1- Assumption: **different groups should have different distributions of the features**
The density of $\underline{X}$ given its label:

$$\underline{X} \mid L = i \sim f_i(\underline{x})$$

And we want $f_i(\underline{x}) \neq f_j(\underline{x})$.
Perform ANOVA/MANOVA to discriminate different groups (reject the hypothesis that $\tau_i - s$ are all equals. We want at least two groups with different distributions)

$f_i(\underline{x})$ is the **probability of observing that feature $\underline{x}$ if you belong to group i.**

2- Prior distribution
Put knowledge of the problem into the model. In a specific environment, some labels are less likely than others.

$$P[L = i] = p_i \qquad i = 1, \dots, p$$

$$p_1, \dots, p_g \geq 0 \quad with \sum_{i=1}^{g} p_i = 1$$

Is **not given by the training set**, but by the setting. Same training set can have different priors. Same features observed in different scenarios have different meaning. (Different priors of Malaria in Italy than priors of Malaria in Kenya).
Is wrong to define the prior based on the training set, since is a limited observation of the problem. Correct to define it on a context dependent basis.

3- Cost of misclassification: how much do I pay if I classify an observation to group i if instead were j?

$$C(i \mid j)$$

Example: $i = patient\ has\ flu \quad j = patient\ has\ covid$

$$C(i \mid j) = the\ patient\ die\ because\ has\ not\ the\ right\ care$$

$$C(j \mid i) = the\ helthcare\ system\ get\ high\ expanses \rightarrow hospital\ \ fails$$

Note**: is not symmetrical.**

Context dependent. Can't be retrieved from training set.
Also, can be different based on the user that is using it: the patient would put a higher cost on $C(i \mid j)$ respect to the director of the hospital, that you put higher cost on $C(j \mid i)$.

Cost of misclassification and priors are related.

Optimal classifier

The optimal classifier is the one that minimize the cost of misclassification.

Classification is the same of partition of the space:

$$\delta: X \rightarrow \{1, \dots, g\}$$

Is equivalent to find partitions of X:

$$R_i = \delta^{-1}(i) = \{\underline{x} \in X : \delta(\underline{x}) = i\} \qquad i = 1, \dots, g$$

Those partitions must:

- have nothing in common, hence void interception:

$$R_i \cap R_j = \emptyset \quad with\ i \neq j$$

- Represent the whole space:

$$\bigcup_{i=1}^{g} R_i = X$$

Optimal classifier criteria

Find δ that minimize the expected cost of misclassification (**ECM**).

We start with a dichotomous classifier (binary classifier).

$$g = 2, \quad so\ we\ want\ to\ find\ R_1, R_2$$

δ is equivalent to partition X into R_1, R_2.

$$\delta(\underline{x}) = 1 \quad iff \quad \underline{x} \in R_1$$
$$\delta(\underline{x}) = 2 \quad iff \quad \underline{x} \in R_2$$

Since is binary: $R_1 = R_2^c$ (complementary).

Remark:

- p_i: probability for the observed unit $\underline{x}$ to belong into group i
- $f_i(\underline{x})$: if $\underline{x}$ belongs to group i, it shows features $f_i(\underline{x})$ with probability $f_i(\underline{x})d\underline{x}$

Therefore: unit belonging to group i:

$$f_i(\underline{x})p_i \, d\underline{x}$$

The misclassification for group 1

➔ according to δ, $\underline{x}$ belongs to R_2, so I integrate (sum of all the costs of misclassification) over R_2.

The same reasoning for the other group.

Expected Cost of Misclassification:

$$ECM[\delta] = \underbrace{\int_{R_2} C(2|1)f_1(\underline{x})p_1 \, d\underline{x}}_{= \int_X C(2|1)f_1(\underline{x})p_1 d\underline{x} - \int_{R_1} C(2|1)f_1(\underline{x})p_1(\underline{x})d\underline{x}} + \int_{R_1} C(1|2)f_2(\underline{x})p_2 \, d\underline{x}$$

This because $R_1 = R_2^c$

$$\int_X C(2|1)f_1(\underline{x})p_1 d\underline{x} = C(2|1)p_1 \quad since\ f_1(\underline{x})\ is\ a\ density\ and\ integrating\ on\ the\ whole\ space = 1$$

$$= C(2|1)p_1 + \int_{R_1} [C(1|2)f_2(\underline{x})p_2 - C(2|1)f_1(\underline{x})p_1]d\underline{x}$$

To minimize the integral, I should choose δ (and so R_1) by minimize the function inside the integral, and so choosing the points into R_1 that minimize the difference inside the integral:

$$R_1 = \{\underline{x} \in R^p : C(1|2)f_2(\underline{x})p_2 \leq C(2|1)f_1(\underline{x})p_1\}$$

For R_2 we know is the complementary of R_1:

$$R_2 = \{\underline{x} \in R^p : C(1|2)f_2(\underline{x})p_2 > C(2|1)f_1(\underline{x})p_1\}$$

➔ This is the optimal classifier.

Generalizing ECM for any g: same argument of binary case but now I can classify as any other group, and I must sum all the possible costs of misclassification.

$$ECM[\delta] = \sum_{k\neq 1} \int_{R_k} C(k|1)f_1(\underline{x})p_1 \, d\underline{x} + \sum_{k\neq 2} \int_{R_k} C(k|2)f_2(\underline{x})p_2 \, d\underline{x} + \cdots + \sum_{k\neq g} \int_{R_k} C(k|g)f_g(\underline{x})p_g \, d\underline{x}$$

We need the integrals for each set, so we must rearrange getting the integrals that are summed for each group inside, and bring them out

$$= \int_{R_1} \sum_{k\neq 1} C(1|k)f_k(\underline{x})p_k \, d\underline{x} + \cdots + \int_{R_g} \sum_{k\neq g} C(g|k)f_k(\underline{x})p_k \, d\underline{x}$$

The optimal classifier assigns the units to the space where the cost of misclassification is lower (doing wrong as low as possible when I'm classifying on that space):

$$R_1 = \{\underline{x} \in R^p : \sum_{k \neq 1} C(1|k) f_k(\underline{x}) p_k \leq \sum_{k \neq 1} C(j|k) f_k(\underline{x}) p_k \quad \forall j \neq 1\}$$
$$R_2 = \{\underline{x} \in R^p : \sum_{k \neq 2} C(2|k) f_k(\underline{x}) p_k \leq \sum_{k \neq 2} C(j|k) f_k(\underline{x}) p_k \quad \forall j \neq 2\}$$
$$\vdots$$

For general i:

$$R_i = \{\underline{x} \in R^p : \sum_{k \neq i} C(i|k) f_k(\underline{x}) p_k \leq \sum_{k \neq i} C(j|k) f_k(\underline{x}) p_k \quad \forall j \neq i\}$$

We'll use **data to estimate the densities $\boldsymbol{f_k(\underline{x})}$**. Once you do that, you can build the optimal classifier (priors and cost are specified by me).

Change notation: i replaced with t. So, we assign $\underline{x}$ to group $t \in \{1, \dots, g\}$. $\delta(\underline{x}) = t$
We assign $\underline{x}$ to group $t \in \{1, \dots, g\}$, (so $\delta(\underline{x}) = t$) if:

$$\sum_{k \neq t} C(t|k) f_k(\underline{x}) p_k \leq \sum_{k \neq j} C(j|k) f_k(\underline{x}) p_k$$

Dividing by $\sum_{j=1}^{g} f_j(\underline{x}) p_j$ to highlight the cost (we are sure that this quantity is greater than 0, since is a probability, and $f_j(\underline{x}) \neq 0$ otherwise we couldn't observe $\underline{x}$):

$$\frac{\sum_{k \neq t} C(t|k) f_k(\underline{x}) p_k}{\sum_{j=1}^{g} f_j(\underline{x}) p_j} \leq \frac{\sum_{k \neq j} C(j|k) f_k(\underline{x}) p_k}{\sum_{j=1}^{g} f_j(\underline{x}) p_j}$$

But what is this term $\frac{f_k(\underline{x}) p_k}{\sum_j f_j(\underline{x}) p_j}$?
The numerator is:

- $\boldsymbol{f_k(\underline{x})}$ the probability that X is what I observe ($X = x$) given that $L = k$. So is the **probability of observing that feature if you belong to group k.**
- p_k is the (**prior**) probability of being in group k
- ➔ $\textit{the numerator is the probability of observing } x \textit{ and being in group } k$

Same reasoning for the denominator, but for every group, so, is the probability of observing $\underline{x}$

$$\frac{f_k(\underline{x}) p_k}{\sum_j f_j(\underline{x}) p_j} = \frac{P[X = \underline{x} \,|L = k] P[L = k]}{\sum_j P[X = \underline{x} \,|L = j] P[L = j]} = \frac{P[X = \underline{x} \,|L = k] P[L = k]}{P[X = \underline{x}]} = \frac{P[X = \underline{x}, L = k]}{P[X = \underline{x}]}$$

And by Bayes:

$$= P[L = k \mid X = \underline{x}]$$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$P[L = k \mid X = \underline{x}] = \frac{P[X = \underline{x} \,|L = k]P[L = k]}{P[X = \underline{x}]}$$

Prior: what is the probability of observing that label (disease) without any observation. $P[L = k]$
Posterior: what is the probability of observing that label (disease) after having observed the realization of the features ($\underline{x}$) (cough: yes, fever: yes). $P[L = k \mid \underline{X} = \underline{x}]$

You are **updating the prior** (the probability a person has disease) **based on the features** you **observe** (**posterior**).

The original equation for the optimal classifier, by another POV:
Optimal classifier in terms of posterior probability

$$\underbrace{\sum_{k \neq t} \overbrace{C(t|k)}^{\text{But you are attributing to t}} \overbrace{P[L = k \,|\underline{X} = \underline{x}]}^{\text{Probability you belong to k}}}_{\text{ECM a posteriori}} \leq \sum_{k \neq j} C(j|k)P[L = k \,|\underline{X} = \underline{x}]$$

$$\sum_{k \neq t} C(t|k) f_k(\underline{x}) p_k \leq \sum_{k \neq j} C(j|k) f_k(\underline{x})\, p_k$$

Indeed, we want to minimize the cost of misclassification with respect to the posterior probability.

Note: cost is invariant to multiplicative factor. Is important to look at ratio of cost ($\frac{c(i|j)}{c(j|i)}$), not on absolute value.

Particular cases:

1- Same cost of misclassification for every group

$$C(i|j) = d$$
$$C(i|i) = 0$$

From the previous:

$$\sum_{k \neq j} d\, P[L = k \,|\underline{X} = \underline{x}] \;= \sum_{k \neq j} d\, P[L = k \,|\underline{X} = \underline{x}]$$

Since d is a constant can be eliminated: $\sum_{k \neq j} P[L = k \,|\underline{X} = \underline{x}] \;= \sum_{k \neq j} P[L = k \,|\underline{X} = \underline{x}]$
That is:

$$1 - P\left[L = t \,\middle|\, \underline{X} = \underline{x}\right] \leq 1 - P[L = j \mid \underline{X} = \underline{x}] \;\; for\; j \neq t$$

$$P\left[L = t \,\middle|\, \underline{X} = \underline{x}\right] \geq P[L = j \mid \underline{X} = \underline{x}] \qquad for\; j = 1, .., g$$

You attribute the unit for which you observe the features x to group t, if the posterior probability of belonging in group t is larger than the posterior probability of belonging in any other group (j).

You observe x, you compute the probability of being diseased with that observation. Then you compute the probability that you're healthy with the same observations. And you classify as the class with the higher probability.

Bayes classifiers are optimal classifiers for this case (**if equals cost of misclassification**).

2- Both cost of misclassification and priors equal for each group.

$$p_i = \frac{1}{g} \qquad \forall i$$

Again, I classify $\underline{x}$ as t if:

$$\frac{f_t(\underline{x})p_t}{\sum_j f_j(\underline{x})p_j} \geq \frac{f_r(\underline{x})p_r}{\sum_j f_j(\underline{x})p_j}$$

I can get rid of the two p_t and p_r since are the same.

And I get:

$$f_t(\underline{x}) \geq f_r(\underline{x}) \quad \forall t \neq r$$

So I classify x to group t if the likelihood of belonging to t is the largest among all the others.

And in this case, the optimal classifier is the **ML classifier** (a special case of Bayes classifiers when priors are the same). This is assuming equal costs of misclassifications and equal priors (not that there aren't). **Very strong assumptions.**

Look plots at end of notes 3-30: the priors have the role of augmenting or diminishing the distributions of the features. So they move the threshold for the classification.

Bayes classifiers are very flexible

If the cost depends only on the true group of the unit misclassified, you can still use Bayes, and is also the optimal classifier.

Classify a diseased patient with fever is the same as classifying with flu: $C(i|j) = c_j$.

$$\sum_{k \neq t} c_k f_k(\underline{x}) p_k \leq \sum_{k \neq j} c_k f_k(\underline{x}) p_k$$

Define $\pi_k = \frac{c_k p_k}{\sum_j c_j p_j}$ $\qquad$ $that\ are\ normalized\ and\ all \geq 0, with\ \sum \pi_k = 1$

$$\sum_{k \neq t} \pi_k f_k(\underline{x}) \leq \sum_{k \neq j} \pi_k f_k(\underline{x})$$

Is a trick to say to the algorithm that the priors are not the p_k but the π_k where you included the cost of misclassifications.

So you can use bayes classifiers with cost of misclassifications with this trick.

See exercise on tablet 30-3

26-3

Bayes classifiers when features are gaussian

Given that I'm in group i, the observation $\underline{X}$ has a distribution that is gaussian:

$$\underline{X} \mid L = i \sim N_p(\underline{\mu}_i, \Sigma_i)$$

The bayes classify $\underline{x}$ to group t if:

$$P\left[L = t \,\middle|\, \underline{X} = \underline{x}\right] \geq P[L = j \mid \underline{X} = \underline{x}] \qquad i = 1, .., g$$

That is the same of:

$$f_t(\underline{x})p_t \geq f_j(\underline{x})p_j$$

We know the densities, that have gaussian distributions

$$\frac{p_t}{\sqrt{(2\pi)^p \det(\Sigma_t)}} \exp\left\{-\frac{1}{2}\left(\underline{x}-\underline{\mu}_t\right)^T \Sigma_t^{-1}\left(\underline{x}-\underline{\mu}_t\right)\right\} \geq \frac{p_j}{\sqrt{(2\pi)^p \det(\Sigma_j)}} \exp\left\{-\frac{1}{2}\left(\underline{x}-\underline{\mu}_j\right)^T \Sigma_j^{-1}\left(\underline{x}-\underline{\mu}_j\right)\right\}$$

Taking log to simplify, you ends up assigning to group t if:

$$\log(p_t) - \frac{1}{2}\log(\det(\Sigma_t)) - \underbrace{\frac{1}{2}\left(\underline{x}-\underline{\mu}_t\right)^T \Sigma_t^{-1}\left(\underline{x}-\underline{\mu}_t\right)}_{d^2_{\Sigma_t^{-1}}(\underline{x},\mu_t)}$$

$$\geq \log(p_j) - \frac{1}{2}\log(\det(\Sigma_j)) - \underbrace{\frac{1}{2}\left(\underline{x}-\underline{\mu}_j\right)^T \Sigma_j^{-1}\left(\underline{x}-\underline{\mu}_j\right)}_{d^2_{\Sigma_j^{-1}}(\underline{x},\underline{\mu}_j)}$$

Note, **if we make the classical assumption for MANOVA**: $p_1 = p_2 = \cdots = p_g$
You attribute to group t if the unit is the closest (in terms of Mahanalobis distance) to the mean μ_t of that group and far away to all other means μ_j.

The classifier partition space of the features into $R_1, R_2, \ldots, R_g$. By diving the whole space into the points that are close to $\mu_1, \mu_2, .., \mu_g$.

$$\delta(\underline{x}) = t \quad IFF \quad d^2_{\Sigma_t^{-1}}\left(\underline{x},\underline{\mu}_t\right) \leq d^2_{\Sigma_j^{-1}}(\underline{x},\underline{\mu}_j)$$

Note: is a strong assumption to state that priors are all the same and all covariance matrixes are equals.

This function has a name:

Quadratic Discriminant Function

$$d_t^Q(\underline{x}) = \log(p_t) - \frac{1}{2}\log(\det(\Sigma_t)) - \frac{1}{2}\left(\underline{x}-\underline{\mu}_t\right)^T \Sigma_t^{-1}\left(\underline{x}-\underline{\mu}_t\right)$$

By bayes classifier, the Quadratic Discriminant Analysis is:

$$\delta(\underline{x}) = t \quad IFF \quad \underline{x} \in R_t = \{\underline{x} \in R^p: \ d_t^Q(\underline{x}) \geq d_j^Q(\underline{x}) \quad \forall j = 1, \ldots, g\}$$

Is a particular case of the Bayes classifier.

Quadratic because of the Mahanalobis term that induce the quadratic form.

By modifying the priors probabilities you enlarge/reduce the quadratic shapes, and so the decision boundaries.

What happens if also $\Sigma_1 = \Sigma_2 = \cdots = \Sigma_g = \Sigma$
Same covariance of features in every group.
The formula ends up removing the log of the determinants:

$$\log(p_t) - \frac{1}{2}\left(\underline{x} - \underline{\mu}_t\right)^T \Sigma^{-1}\left(\underline{x} - \underline{\mu}_t\right) \geq \log(p_j) - \frac{1}{2}\left(\underline{x} - \underline{\mu}_j\right)^T \Sigma^{-1}\left(\underline{x} - \underline{\mu}_j\right)$$

And survives only the products with x, Σ and μ_j and μ_j:

$$\log(p_t) + \underline{x}^T \Sigma \underline{\mu}_t - \frac{1}{2}\mu_t^T \Sigma^{-1} \underline{\mu}_t \geq \log(p_j) + \underline{x}^T \Sigma \underline{\mu}_j - \frac{1}{2}\mu_j^T \Sigma^{-1} \underline{\mu}_j$$

I removed the quadratic part, is a linear function of x: linear boundaries.

Linear Discriminant Function

$$d_t(\underline{x}) = \log(p_t) + \underline{x}^T \Sigma \underline{\mu}_t - \frac{1}{2}\mu_t^T \Sigma^{-1} \underline{\mu}_t$$

Here I only need one covariance matrix, that is common to all groups.

If you expect curve boundaries between features, use QDA.

LDA is very robust to the assumption of Gaussianity. Is very general, even for complex tasks, by changing the problem a little to make it usable.
Moreover, has a better generalization, hence, less overfitting.

Up to now we always used models. How can we use also training set (so the data)?

Use training data to estimate the distribution of X in the groups

So use the data to estimate f_i.
In another language: $\underline{X} \mid L = i$

In LDA we must estimate from data:

- $\underline{\mu}_1, \underline{\mu}_2, \dots, \underline{\mu}_g \; and \; \Sigma$

In QDA we must estimate from data:

- $\underline{\mu}_1, \underline{\mu}_2, \dots, \underline{\mu}_g \; and \; \Sigma_1, \Sigma_2, \dots, \Sigma_g$

In both:

- $\underline{\mu}_i$ is estimated by $\overline{\underline{X}}_i = \frac{1}{n_i}\sum_{\{j:l_j=i\}} \underline{X}_j$

In QDA $\Sigma_1, \Sigma_2, \dots, \Sigma_g$ estimated by:

$$S_i = \frac{1}{n_i - 1} \sum_{\{j:l_j=i\}} (\underline{X}_j - \overline{\underline{X}}_i)(\underline{X}_j - \overline{\underline{X}}_i)\text{^}T$$

QDA requirement: lots of sample in each group to estimate the covariance. $(\boldsymbol{n >> p})$.

In LDA I only need lots of samples in general, not in every group, to estimate:

$$S_{pooled} = \frac{1}{n - g} \sum_{\{j:l_j=i\}} (n-1) S_j$$

Naïve Bayes

Parametrize (don't consider all the possible matrices, since you must compute the inverse, that is expensive and may can't be find if the matrix is sparse (low amount of data)) $\Sigma_1, \Sigma_2, \dots, \Sigma_g$ by assuming that each of them is diagonal. **Assume independent components** (since we're in gaussian world)**: strong assumption.**

$$d_t^Q(\underline{x}) = \log(p_t) - \frac{1}{2}\sum_{i=1}^{p}\log(\sigma_{ii}^k) - \frac{1}{2}\sum_{i=1}^{p}\underbrace{\frac{(x_i - \overline{x}_{ti})^2}{\sigma_{ii}^t}}_{Standardize\ the\ components\ of\ groups}$$

With $\overline{x}_{ti}$ being i-th component of $\underline{\overline{x}}_t$

Component (feature) wise estimate the variances:

$$\sigma_{ii}^t = \frac{1}{n_t - 1}\sum_{\{j:l_j=t\}}(x_{ji} - \overline{x}_{ti})^2$$

You only need at least 2 observations in group t: $n_t - 1 > 0$. With this amount you compute mean and variance for the group….is garbage. No covariance estimated…
Too simple, and in fact doesn't work.

An even simpler classifier: KNN

KNN

For a k you choose:

$$N_K(\underline{x}) = \{closest\ k\ units\ in\ the\ training\ data\ (to\ \underline{x}, that\ is\ the\ point\ I\ want\ to\ classify)\}$$

So $\delta(\underline{x}) = l$ means that l is the label shown by the majority of the units $\underline{x_i}$ in the training set belonging to $N_K(4)$.
Doesn't work for the curse of dimensionality. Also, close in what sense?

It works only if you cook up a good distance for the problem and the dimensionality of the problem is not big. A solution could be to reduce the dimensionality before.

The most complex model is KNN with $k = 1$. The bigger the k, the less complex the model, the lower the variance, the higher the bias.
$k = 1$ means to classify a unit as the closest point I have.

Fisher argument for LDA

We'll arrive to LDA **without assuming Gaussianity**. That means that Gaussianity is not that important for LDA!

We'll show:

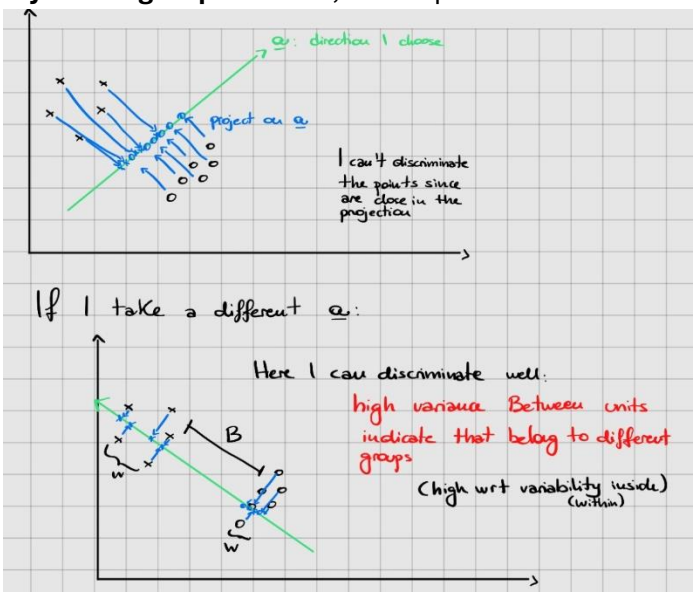
- Robustness of LDA to Gaussianity assumption
- Bring dimensionality reduction suitable for the classification problem (so reduced space, but showing how is the classification performed, on which direction)

Let's find the direction that maximize the discrimination between the groups. Hence, the variability between groups on the projected space on the direction I choose, must be much higher than the variability within groups.

Fisher's idea: **transform the multivariate observations into a univariate observation**, that is the projection of the original multivariate into a direction. We want to have a direction such that in the univariate space the separations of the points from a population are well separated from point of the others.

$\underline{X} \mid L = i \sim \left(\underline{\mu}_i, \Sigma\right)$ *that is a general distribution*, with **same covariance in every group**

I assume same variability within groups: indeed, uses a pooled estimate.



Variability between groups is estimated using the means of the groups, compared to the overall means:

$$B = \frac{1}{g-1}\sum_{i=1}^{g}\left(\underline{\mu}_i - \overline{\underline{\mu}}\right)\left(\underline{\mu}_i - \overline{\underline{\mu}}\right)^T$$

$$\overline{\underline{\mu}} = \frac{1}{g}\sum_{i=1}^{g}\underline{\mu}_i$$

I want to compare the variability between groups with the variability within groups.

For a generic linear combination of the original observations:
$\underline{a} \in R^p$ and $\underline{X} \mid L = i \sim \left(\underline{\mu}_i, \Sigma\right)$

We want find the vector that identifies the separations between the classes.

The expected value and the variance if $L = i$:

$$E\left[\underline{a}^T \underline{X} \mid L = i\right] = \underline{a}^T \underline{\mu}_i$$
$$Var\left[\underline{a}^T \underline{X} \mid L = i\right] = \underline{a}^T \Sigma \underline{a} \ \ doesn't\ depend\ on\ i$$

The variability within groups, that is assumed the same for every group, is $a^T \Sigma a$. And is our meter. We want to find the direction a that maximizes the ratio of the variability between groups and the variability within groups:

$$\arg\max_{\underline{a}} \frac{\underline{a}^T B \underline{a}}{\underline{a}^T \Sigma \underline{a}} = \ arg\max_{\underline{a}} \frac{1}{g-1} \sum_{i=1}^{g} \frac{\left(\underline{a}^T \underline{\mu}_i - \ \underline{a}^T \overline{\underline{\mu}}\right)^2}{\underline{a}^T \Sigma \underline{a}}$$

Let's transform the variables:

$$\underline{u} = \ \Sigma^{1/2} \underline{a}$$

And so:

$$\underline{a} = \ \Sigma^{-1/2} \underline{u}$$

The original problem becomes:

$$\frac{\underline{u}^T \Sigma^{-1/2} B \Sigma^{-1/2} \underline{u}}{\underline{u}^T \underline{u}}$$

And now I must find the $\underline{u}$ that:

$$arg\max_{\underline{u}} \frac{\underline{u}^T \ \Sigma^{-1/2} \ B \ \Sigma^{-1/2} \ \underline{u}}{\underline{u}^T \underline{u}}$$

The $\underline{u}$ that maximize this, is the same found with the corollary used for the PCA. By finding the spectral decomposition of $\Sigma^{-1/2} \ B \ \Sigma^{-1/2}$:

$$\Sigma^{-1/2} \ B \ \Sigma^{-1/2} = \ \sum_{i=1}^{p} \lambda_i \underline{e}_i \underline{e}_i^T$$

And therefore, the argmax is the eigenvector corresponding to the largest eigenvalue:

$$arg\max_{\underline{u}} \frac{\underline{u}^T \ \Sigma^{-1/2} \ B \ \Sigma^{-1/2} \ \underline{u}}{\underline{u}^T \underline{u}} = \frac{\underline{a}^T B \underline{a}}{\underline{a}^T \Sigma \underline{a}} = \underline{e}_1$$

Hence, for the change of variable:

$$\underline{a} = \ \Sigma^{-1/2} \underline{u} \ \ \rightarrow \ \ \underline{a} = \ \Sigma^{-1/2} \underline{e}_1$$

$$\underline{a}_1 = \ \Sigma^{-1/2} \underline{e}_1$$
$$\underline{a}_2 = \ \Sigma^{-1/2} \underline{e}_2$$
$$\vdots$$
$$up\ to\ how\ many?$$

How many eigenvalues different from zero has that matrix?
Σ has rank p: is the covariance generated by the means inside a group, so the number of features.
B has rank $g - 1$ since is the covariance generated by the means of the groups.

And so, the rank of the overall matrix, is: $\min(g-1, p) = s$
So we have s eigenvalues/eigenvectors.
Once I find the direction $\underline{a}$, every vector $\underline{X}$ is transformed in new coordinates, that are the projections on those directions:

$$\underline{a}_1^T \underline{x}, \ldots\ldots, \underline{a}_s^T \underline{x}$$

Those are called **Fisher Discriminant Scores.**

If I compute the covariance between two of those scores:

$$Cov(\underline{a}_i^T \underline{x}, \underline{a}_j^T \underline{x}) = \underline{a}_i^T \Sigma \underline{a}_j = \underline{e}_i^T \Sigma^{-1/2} \Sigma^{-1/2} \underline{e}_j = \underline{e}_i^T \underline{e}_j = \begin{cases} 0 \; if \; i \neq j \\ 1 \; if \; i = j \end{cases}$$

The don't depend on where I am, since assuming same Σ in every group, and can only be 0 or 1.

So if we put all the directions into a matrix A representing all possible transformations:

$$Cov(A\underline{x}) = I$$

I'm transforming my data into a new space, where the metric is the Euclidean one! You can say **here** that two points are close if they are close in the screen/paper.

The scores are ordered, so you may consider only some scores, exactly like principal components.
But here you look for the eigenvectors of $\Sigma^{-1/2} \; B \; \Sigma^{-1/2}$.

How to build the classifier

I have to estimate:

- $\underline{\mu}_i$ estimated by $\overline{\underline{x}}_i$
- Σ estimated by $\mathrm{S}_{\mathrm{pooled}}$

$\mathrm{S}_{\mathrm{pooled}} = \frac{W}{n-g}$ so we need to find the within variability.
That is: $W = \sum_{i=1}^{g}(n_i - 1)S_i = \sum_{i=1}^{g}\sum_{j=1}^{n_i}(\underline{x}_{ij} - \overline{x}_i)(\underline{x}_{ij} - \overline{x}_i)^T$

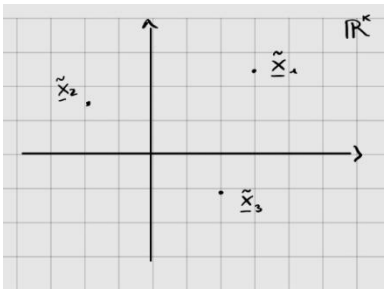
To find the $\underline{a}$ I must find the eigenvectors of $S_{pooled}^{-1/2} \; \hat{B} \; S_{pooled}^{-1/2}$.

Where $\hat{B} = \frac{1}{g-1}\sum_{i=1}^{g}(\overline{x}_i - \overline{\underline{x}})(\overline{x}_i - \overline{x})^T$
With $\overline{\underline{x}} = \frac{1}{g}\sum \overline{x}_i$ and $\overline{x}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} \underline{x}_{ij}$

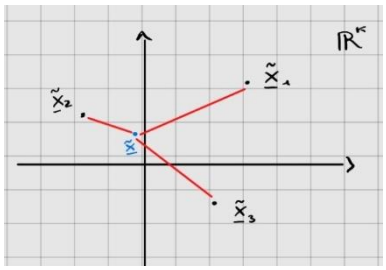
Take $\overline{\underline{x}}_i$ and transform it considering its Fisher scores:

$$\tilde{\underline{x}}_i = (\underline{a}_1^T \overline{\underline{x}}_i, \underline{a}_2^T \overline{\underline{x}}_i, \ldots, \underline{a}_k^T \overline{\underline{x}}_i) \qquad with \; k \leq s$$

I've transformed the means and projected into R^k, so I have:



And how do I classify? How is $\delta(\underline{x})$ made? I need to attribute to $\underline{x}$ the group where it belongs to.
So I transform the new observation $\underline{x}$ into $\tilde{\underline{x}}$, you measure with **Euclidean distance** to which group mean is the closest, and you attribute to it.



So I attribute to group t if I'm closer to the mean of group t with respect to the projection of any other mean. This means that we're splitting the plane into regions (the red lines in the plot.)

This classifier is exactly the same as LDA if priors are all the same, but without Gaussianity assumption.

This proved that LDA is robust to Gaussianity, and that **we have a way to perform dimensional reduction consistent to discrimination.**
You typically use any classifier you want and then to represent the data and the classification you made on the Fisher scores, since is the best perspective you can have.

4-4

To evaluate how good is the classifier we estimate the Actual Error Rate: is counting the errors made by classifying 2 as 1, 3 as 1, 4 as 1 and so on.

$$AER = \sum_{k\neq 1}\int_{R_k} f_1(\underline{x})p_1\, d\underline{x} + \sum_{k\neq 2}\int_{R_k} f_2(\underline{x})p_2\, d\underline{x} + \cdots + \sum_{k\neq g}\int_{R_k} f_g(\underline{x})p_g\, d\underline{x}$$

Is summing all the errors you're making on each partition.
If you knew the density functions and the priors, you could train the classifier to minimize this error.
Unfortunately, we don't know them.

How to estimate AER with the training set

A naïve idea is to see what δ perform on the dataset (training set) and count the mistakes.
Fix an example for 2 groups. Say that n_1 is the total true unit of group 1, and n_2 is the total true unit of group 2.
n_{11}: true unit of group 1 classified as group 1
n_{12}: true unit of group 1 classified as group 2 (error)
n_{21}: true unit of group 2 classified as group 1(error)
n_{22}: true unit of group 2 classified as group 2

So the error rate **estimated** is $\frac{n_{12}+n_{21}}{n}$ and is called **APER: Apparent Error Rate**.

$\boldsymbol{APER = \widehat{AER}}$

If we decompose, we can see:

$$\frac{n_{12}}{n} + \frac{n_{21}}{n} = \underbrace{\frac{n_{12}}{n_1}}_{estimate\ \int_{R_2} f_1(x)dx}\ \underbrace{\frac{n_1}{n}}_{estimate\ p_1} + \frac{n_{21}}{n_2}\frac{n_2}{n}$$

Is too optimistic. Since I'm using the training data.

It works only if the training data is sampled randomly in the population, that is very hard…imagine to classify rare diseases, you should maintain original percentages. And in this way, for the classifier is hard to learn with such low percentage of samples of diseases.

So this is a very simple estimate for AER, but doesn't really gives a metric to evaluate classifiers.

Observation for dichotomous classifiers

$$\delta(\underline{x}) = \begin{cases} 1 & if\ positive \\ 0 & if\ negative \end{cases}$$

$$n_{10}: false\ negative$$
$$n_{01}: false\ positive$$

$$APER = \frac{n_{10}+n_{01}}{n}$$

Precision: amount of positive value predicted correctly

$$\frac{n_{11}}{n_{*1}}$$

Recall: $\frac{n_{11}}{n_{1*}}$: $(\ also\ called\ Sensitivity)$

Specificity: $\frac{n_{00}}{n_{*0}}$

- If precision is high, the false positives are low
- If recall is high, the false negatives are low

All those indexes (and others) depend on the estimate of the AER, since are calculated starting from the confusion matrix. So if you estimate in the wrong way the AER, they will be not realistic.

A more realistic estimate of AER (and the above indexes)

By using **test data** to calculate the errors**.**
After having trained δ with the train data ($X \in R^m$), calculate with some test data $\tilde{x}$:

$$\widehat{AER} = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}[\delta(\tilde{x}_i) \neq \tilde{l}_i]$$

That is the average errors that is making with new data.
In this way you don't enforce the data to have a good classifier.
However you'd like to use all the data you have to train.

Cross validation

Using cross validation, you use training and test data iteratively.

Leave One Out Cross Validation: take a unit i out from the whole dataset and train the classifier on the new dataset. Then evaluate the classifier trained without that unit. How do you evaluate? With the unit leaved out and you see how it classify. $\varepsilon_i = \begin{cases} 1 & if\ \delta_{-1}(x_i) \neq l_i \\ 0 & if\ \delta_{-1}(x_i) = l_i \end{cases}$
You repeat this procedure for each unit of the dataset, and you count how many data points you classified wrongly at testing time. And you take the average of misclassification done:

$$\widehat{AER} = \frac{1}{n}\sum_{i=1}^{n} \varepsilon_i$$

At the end of the procedure, you train with the whole training set.
Therefore, you have **low bias** but high variability for **estimating the AER**.

K-fold Cross Validation: aim to reduce the variability, by paying some bias. (Remark: mean squared error is the sum of bias and variance). The classifiers are less dependent, and I'm using less classifiers (less variance).
Split the training set into k subsets. Is important that this split is performed on a random order. The single subsets must be representative of the whole dataset. (Shuffle data first). Moreover, the subsets must be balanced.

Perform training on the dataset without one of the k subsets and evaluate with the data leaved out.

You evaluate the errors on each single unit:

$$\varepsilon_{ij} = \begin{cases} 1 & if\ \delta_{partition_i}(x_j) \neq l_j \\ 0 & otherwise \end{cases}$$

The error on the unit j:

$$Err_i = \frac{1}{n_i} \sum_{x_j \in partition_i} \varepsilon_{ij}$$

$$AER = \frac{1}{n} \sum_{i=1}^{k} n_i Err_i$$

Note: you can use the indices of precision, recall and so on with Cross Validation.

How to choose k?

If $k = n$ is the same of Leave One Out. Therefore, I want $k << n$, and also a number for which n can be divided with: $n\%k == 0$ in this way you have equally sized chunks.

- If the

A very accurate strategy to estimate AER: repeat k-fold several times

K fold can be repeated again, shuffling data one more time after a finish! In this way you create new partitions, training the classifiers with new chunks and assessing their performance in a more general environment.

So repeat k fold cross validation B times, every time shuffling data. In this way you get:

$$\widehat{AER}_1, \widehat{AER}_2, \dots, \widehat{AER}_B$$

You have B estimate, so you can look at them as a distribution of you sample, and compute the mean and variance:

- $E\left[\widehat{AER}(\delta)\right] = \widehat{AER_m} = \frac{1}{B}\sum_{i=1}^{B} \widehat{AER}_i$
- $Var\left[\widehat{AER}(\delta)\right] = \frac{1}{B-1} \sum_{i=1}^{B} (\widehat{AER}_i - \widehat{AER}_m)\text{^}2$

In this way you have an idea of the distribution of the actual error rate. You can change the k in those B times

Is a bootstrap way to get the distribution of that estimate of Actual Error Rate.

$$Var\left(\frac{1}{n}\sum_{i=1}^{n} \varepsilon_i\right) \ll Var(\varepsilon_i)$$

With leave one out cross validation, **all the** (lots of) **classifiers** I build, **are** almost **correlated**, since the only difference there is, is a single data point. Too much dependence among classifiers, and for this reason low bias but high variance.

With k-fold, you generate **k classifiers, closer to be independent**, therefore, lower variance than LOO.

8-4

Support Vector Machine

Main characteristic: robust, but prone to overfit.

Aims at **binary supervised classification problems**.
We have our dataset composed of $\underline{x_i} \in R^p$ and labels $l_i \in \{-1,1\}$.

Find an hyperplane that separates R^p in R_1 and R_2, with R_1 being the subset containing the $\underline{x_i}$ with label l_i
The new thing is not in the linearity of the separation, but in how this hyperplane is found.

But when does an hyperplane exists that separates the two classes?
More formally: given $A, B\ included\ in\ R^p$ when can they be separated by an hyperplane?

Assumption: $Let\ CH(A)\ and\ CH(B)$ be the smallest convex set containing A and the smallest convex set containing B (called Convex Hull).
If:

1) Both the convex hulls aren't empty $CH(A) \neq \emptyset\ and\ CH(B) \neq \emptyset$
2) $CH(A) \cap CH(B) \neq \emptyset$
3) Either:
    a. $CH(A)\ or\ CH(B)\ is\ open$
    b. $CH(A)\ and\ CH(B)$ are closed and at least one of them is compact (you can cover the set with a finite number of close set)

$\rightarrow$ $the\ separating\ hyperplane\ exists$: at a side there are all elements of A and on the other side all the elements of B

In our case: 3b is always satisfied, since we're always dealing with finite number of points in A, B.
You can indeed close the convex hull.
So what we really need is first and second condition.

However, there are multiple separating hyperplanes in general case (see the simplest classification plot of dots and cross). Which one do we choose?
We must solve an optimization problem, but first some definitions.

An **hyperplane** $L\ on\ R^p$ is an affine subspace of R^p with dimension $p-1$.

Let $\beta \in R^p$ a direction vector, so with norm 1, orthogonal to the hyperplane L.
Let $\underline{x}_0$ be the intersection of the linear space on the direction $\underline{\beta}$ and the hyperplane L:

$$\underline{x}_0 \in span\left(\underline{\beta}\right) \cap L$$

Define $\beta_0 = ||\underline{x}_0||$, so the norm of x_0.
A generic point $\underline{x}$ belongs to the hyperplane $L \leftrightarrow$ projecting $\underline{x}$ into β is equal $\underline{x}_0$:

$$\underline{x} \in L \quad IFF \quad \pi_{\underline{x}|\beta} = \underline{x}_0$$

That is:

$$\frac{\underline{\beta}\underline{\beta}^T}{\underline{\beta}^T\underline{\beta}}\underline{x} = \underline{x}_0$$

But since $\underline{\beta}$ has unitary norm:

$$\underline{x} \in L \quad IFF \quad \underline{\beta}^T\underline{x} = \beta_0$$

In fact, is the equation of the hyperplane.
(See identify point on hyperplane on tablet)

So a point not belonging to the hyperplane have score either $\underline{\beta}^T\underline{x} > \beta_0$ or $\underline{\beta}^T\underline{x} < \beta_0$.
So if $\underline{\beta}^T\underline{x} - \beta_0 > 0$ we classify x with label +1, otherwise with -1.

Notice that $\underline{\beta}^T\underline{x} - \beta_0$ is the distance of $\underline{x}$ from the linear space.

So we have a score of distance from our point to the hyperplane. Therefore, our hyperplane defines a classifier:

$$Let\ y_i = \begin{cases} +1\ if\ unit\ i\ belongs\ to\ Class\ 1 \\ -1\ if\ unit\ i\ belongs\ to\ Class\ 2 \end{cases}$$

With this trick of +1/-1 I can use a better notation:

$y_i\left(\underline{\beta}^T\underline{x_i} - \beta_0\right)$ is the distance between $\underline{x_i}$ and L

And now, $y_i\left(\underline{\beta}^T\underline{x_i} - \beta_0\right) \geq 0$ always.

L is defined by both $\underline{\beta}$ and β_0: the vector and the point.

I want to find the region of space (a slab) where there are no points, of both the classes.
The greater the slab, the more the classes are separated, so I want to find the hyperplane that maximize the slab.
The slab is found by finding the closest points of the two classes from the hyperplane.

We call **margin** M (or slab) the minimum distance to the hyperplane:

$$M = \min\left\{y_i\left(\underline{\beta}^T\underline{x_i} - \beta_0\right), \forall\ i = 1, \dots, n\right\}$$

We aim to find the hyperplane (vector and point) with largest margin:

$$\max_{\beta,\beta_0} M$$

$$\text{St:} \begin{cases} ||\beta|| = 1 \\ y_i\left(\underline{\beta}^T\underline{x_i} - \beta_0\right) \geq M\ \forall i \end{cases}$$

The solution is:

$$\underline{\hat{\beta}} = \sum_{i=1}^{n} \lambda_i\, y_i \underline{x_i}$$

Where λ_i are coefficient from Lagrangian multiplier.

The function that defines the hyperplane will be:

$$f(x) = \hat{\underline{\beta}}^T \underline{x} - \beta_0 = \sum_{i=1}^{n} \lambda_i \, y_i \underbrace{\underline{x_i}\underline{x}}_{\substack{Only\ inner\ product \\ of\ the \\ point\ to\ classify\ is\ used!}} - \hat{\beta}_0$$

Classification:

$$C(\underline{x}) = Sign(f(\underline{x}))$$

The important points are the one close to the hyperplane (that are called support vectors). Important in the sense that they're the only one considered to choose the hyperplane.
Is a major difference respect to the LDA/QDA: there we used all the points to estimate covariance between and within. Also points very far from the hyperplanes are considered equally important. Here we're not!
For this reason, is **more robust to outliers** since **consider few points to assess the boundary**.

Cons: **no priors** used, is a purely geometric classification.

But this is for overly simple cases, when the two classes are well separated in the plane.
When the hyperplane doesn't split exactly the two classes, because there isn't a clear separation between class, there are two ideas:

1- Allow some point to cross the lines: we don't want all the units in a region and all the others on the other region, but we relax.
2- Exploit the curse of dimensionality: transform the data into a multidimensional (larger) space. We want the points to be very well separated (and in curse of dimensionality they are), and so, the hyperplane should very well separate the points.

Idea 1: relax the optimization problem

Instead of imposing the minimal distance from a point to the hyperplane, $y_i \left(\underline{\beta}^T \underline{x_i} - \beta_0\right) \geq M$, we relax:

$$y_i \left(\underline{\beta}^T \underline{x_i} - \beta_0\right) \geq M(1 - \varepsilon_i)$$

With $\varepsilon_i \geq 0$ and $\sum \varepsilon_i \leq C$ (Budget constraint, if C=0 is equivalent to the hard problem).
If C is very large, any hyperplane would satisfy the optimization problem.
So we're asking that the distance can be smaller from the hyperplane.

Idea 2: exploit curse of dimensionality

Take an egg with dots as the yolk and cross as the white around. If you get a new set of coordinates, centring the new reference system in the middle, using polar coordinates, you end up having a linear separation between the two points.
Polar coordinates is just an example. You have to find the transformation that create the new variables in a well separated form.

Idea 2.b: kernel methods

Take a set of functions from $R^p \rightarrow R$:

$$h_1, \dots, h_m \qquad with \quad h_i: R^p \to R$$

And let $h(\underline{x}) = \left(h_1(\underline{x}), \dots, h_m(\underline{x})\right) \in R^m$
You can use any function you want, that must be helpful to separate the points.
The more functions you take (the bigger the m), the bigger is the new space, the more curse of dimensionality -> the more you can exploit separations between the transformed variables.

The transformed dataset becomes: $(h(\underline{x}_1), h(\underline{x}_2), \dots, h(\underline{x}_n))$.

In this transformed space you can find either an hard or soft separation hyperplane!

The new solution is:

$$\hat{\beta} = \sum_{i=1}^{n} \lambda_i \, y_i h(x_i)$$

So:

$$f(x) = \underline{\hat{\beta}}^T h(\underline{x}) - \beta_0 = \sum_{i=1}^{n} \lambda_i \, y_i \underbrace{h^T(\underline{x}_i) h(\underline{x})}_{<h(\underline{x}_i), h(\underline{x})>} - \hat{\beta}_0$$

The idea becomes, instead of looking at all possible transformations h-s, find $k : R^p \; x \; R^p \to R$
st $k(\underline{x}, \underline{w}) =< h(\underline{x}), h(\underline{w}) >= h^T(\underline{x}) h(\underline{w}) \; \forall \underline{x}, \underline{w}$.

This function defines how the inner product is.
If you believe that such a function exists, the solution is:
$f(x) = \sum_{i=1}^{n} \lambda_i \, y_i k(x_i, x)$
That is way easier computationally: you already know the form of the solution.

Some kernels:

- $k(\underline{x}, \underline{w}) = \left[1 + \underline{x}^T \underline{w}\right]^d$
- $k(\underline{x}, \underline{w}) = exp\left[-\gamma \left|\left|\underline{x} - \underline{w}\right|\right|\right]^2$

The risk is that you enlarge too much the space, to exploit more the curse of dimensionality, and you find some clear separations of the data, but you end up overfitting since the separation is due to the enlargement. Low generalization. You end up doing the same thing of a KNN with k=1, so repeating the training set.
Using cross validation, you can keep under control the budget constraint and generalization error.

Cluster analysis (unsupervised classification)

You believe that different classes of units exist, but you can't observe the labels of the training set.

Goal:

- Estimate g (how many groups there are)
- Estimate the labels $(l_1, \dots, l_n)$

After that you end up with supervised method with your estimated groups.

Idea: units that belong to the same group are more similar that units belonging to different groups.

We must define, however, what do we mean by "similar". We must define a function that tells us if you things are similar. The naïve approach we'd have is to say they are similar if they are close…however, two similar object can be distant in some metric and very similar to some other metric, more appropriate for the problem.

Dissimilarity definition

The higher the value given by the function, the higher the dissimilarity between x and y.

$$d(x, y)$$
$$d: R^p \; x \; R^p \rightarrow R$$

Properties of this function.

1- $d(\underline{x}, \underline{x}) = 0 \quad \forall \underline{x} \in R^p$
   a. A strongly way: if two elements have dissimilarity 0, those two elements are the same: $d\left(\underline{x}, \underline{y}\right) = 0$ iff $\underline{x} = \underline{y}$
2- **Symmetry**: $d\left(\underline{x}, \underline{y}\right) = d\left(\underline{y}, \underline{x}\right)$
3- **Triangular inequality:** $d\left(\underline{x}, \underline{y}\right) \leq d(\underline{x}, \underline{z}) + d\left(\underline{z}, \underline{y}\right) \quad \forall \underline{x}, \underline{y}, \underline{z} \in R^p$
4- $d\left(\underline{x}, \underline{y}\right) \leq \max\left\{d(\underline{x}, \underline{z}), d\left(\underline{z}, \underline{y}\right)\right\} \; \forall \underline{x}, \underline{y}, \underline{z} \in R^p$

Metric definition: satisfy $1, 2, 3$.
Pseudo metric definition: satisfy $1, 2$.
Ultra metric definition: satisfy $1,2,4$. Note that if satisfy 4 also satisfy 3.

16-04

Cluster analysis

The similarity function can be very general, for instance, if our units are sequences of characters (the genome), you can specify the similarity based on the position of characters. For example, say n is the first character when the two sequences start to differ, and define $d\left(\underline{x}, \underline{y}\right) = \frac{1}{|\underline{x}|^n}$. The more the two sequences are similar at the start of the sequences, the lower is the dissimilarity (since n is greater).

A list of general purpose distances

Euclidean distance: $d\left(\underline{x}, \underline{y}\right) = \sqrt{\left(\underline{x} - \underline{y}\right)^T \left(\underline{x} - \underline{y}\right)}$

Typically applied to standardized data (both $\underline{x}$ *and* $\underline{y}$), since you don't want to consider variability in distance.

Euclidean distance won't cluster based on the shape and trends of the variables.

Dissimilarity based on shape of functions

If you want to capture the shape of functions, take the angle between them, that is the correlation:

$$d\left(\underline{x},\underline{y}\right) = \sqrt{2(1 - Corr\left(\underline{x},\underline{y}\right)}$$

Mahanalobis distance: $d\left(\underline{x},\underline{y}\right) = \sqrt{\left(\underline{x}-\underline{y}\right)^T \Sigma \left(\underline{x}-\underline{y}\right)}$

Use it **if you have a clear idea of the variability** of the components of the vectors describing the units. The issue of this distance is that, in cluster analysis **typically you don't know the covariability between the clusters**.

Minkowski distance: $d\left(\underline{x},\underline{y}\right) = \left(\sum_{i=1}^{p} \left|\underline{x}_i - \underline{y}_i\right|^r\right)^{1/r}$

- r= 2: is equivalent to Euclidean distance
- r=1: **Manhattan distance**

The higher the r, the higher the weight you give to the large difference between the components.

Canberra distance: $d\left(\underline{x},\underline{y}\right) = \sum_{i=1}^{p} \frac{\left|\underline{x}_i - \underline{y}_i\right|}{\left|\underline{x}_i\right| + \left|\underline{y}_i\right|}$

Compare the difference with the average, to see how the difference is relevant with respect to the phenomena. Is a relative difference.
Very sensitive to small changes when both are close to 0.

Dissimilarity on categorical variables

Euclidean matches for discordancy between categorical variables. So count how many times the two vectors differ.
X = [0 0 0 1 1 0], y = [0 0 1 1 1 0]
With Euclidean distance you get d(x,y)=1. That are the couple of components that are different.
Percentage of dissimilarity.
The issue in this is that is not weighting in any way "low dissimilar" units, treating them as different as "hard dissimilar" elements.

But what if you have both categorical and quantitative variables?
You can describe two distances, one for the categorical and one for the quantitative variables, or even a convex combination of those distances.

You can cluster also not the units, but the features! Again, could be a good idea to look for the correlation between the features. They have high correlation if are similar apart from translation.

Once d is specified, you must compute the matrix D that contains all the couples of dissimilarities between each unit.

$$D_{ij} = d(\underline{x}_i, \underline{x}_j)$$

It has null diagonal terms and is symmetric.

Most of the cluster algorithms starts directly from the dissimilarity matrix, without even analysing the original data.

We need also to specify the distance between clusters.

Distance between subsets of R^p

We need to define a distance between the clusters, to understand when to merge them.
Remark: we have finite subsets.

$$Let\ U, V\ be\ finite\ subsets\ of\ R^p$$

How can we define $d(U,V)$?
Multiple options:

- **Single linkage:** $d(U,V) = \min\{d\left(\underline{x}_i, \underline{y}_i\right),\ \ \underline{x} \in U, \underline{y} \in V\}$
  Here you want to see the two elements of the subsets that are more similar.
- **Complete linkage:** $d(U,V) = \max\{d\left(\underline{x}_i, \underline{y}_i\right),\ \ \underline{x} \in U, \underline{y} \in V\}$
- **Average linkage:** $d(U,V) = \frac{1}{\#U\ \#V}\sum_{\underline{x}_i \in U}\sum_{\underline{y}_i \in U} d(\underline{x}_i, \underline{y}_i)$
- **Centroid linkage:** you calculate the distance between the centroid of the two clusters.
  If you use Euclidean distance, the centroid is typically:
  - **Baricenter of the dataset** (can be a point not of the dataset): can be relaxed to be the point in the dataset that minimize the distance with other units.

The distance that you use between clusters must be consistent with the distance used between units.

Single linkage between Italy and Switzerland is 0.
Complete linkage between Italy and Switzerland is around 2000km.

Hierarchical agglomeration clustering algorithm

The opposite of splitting clusters.

1) *Every unit is a cluster*
2) *Calculate the initial D matrix between each cluster*
3) *Merge the two closest clusters (in terms of cluster distance)*

4) $Update\ D\ and\ store\ wich\ clusters\ have\ been\ merged$
5) $Repeat\ 3-4\ until\ there\ is\ only\ a\ cluster$

The dendrogram show this process: starting from the bottom, where every unit is a cluster, at a certain height (that is the dissimilarity of the joined clusters) the joining of the clusters.
To create clusters, you must specify at which distance (with which lens definition to look the data) you want to see the clusters.

What if several clusters have same distance? Which one do I cluster?

➔ Jittering before.

Double advantage of jittering: you know how robust the cluster structure that you created is.

Cophenetic distance

Is the distance at the time of the join between two clusters in the final clustering structure.
Is an **ultra-metric**.
In other words:

- Is the height of the dendrogram where the two branches that include the two objects (that you want to measure the cophenetic distance) merge.

You want to see if the cophenetic metric is close to the one you used at the beginning.
If the cophenetic distance are all the same, then all the clusters are correlated, hence no really sense to clusterize.

18-04

Ward's method for hierarchical clustering

The idea is to explore the variability within each cluster, treating it as a cost function.

Considering the Euclidean distance $d\left(\underline{x},\underline{y}\right) = \sqrt{\left(\underline{x}-\underline{y}\right)^T\left(\underline{x}-\underline{y}\right)}$, and let C be a set of points in R^p (a cluster).

With d Euclidean distance, we can easily calculate the **barycentre**:

$$\overline{\underline{X}} = arg\min_{\underline{x}} \sum_{\underline{x}_i \in C} d^2(\underline{x}_i,\underline{x}) = \frac{1}{|C|}\sum_{\underline{x}_i \in C}\underline{x}_i$$

For a general distance (fix Euclidean in this case), for each cluster i, we call it ESS_i:

$$ESS_i = \sum_{\underline{x}_j \in C_i} d^2\left(\overline{\underline{x}}_i,\underline{x}_j\right) = \sum_{\underline{x}_j \in C_i}\left(\overline{\underline{x}}_i-\underline{x}_j\right)^T\left(\overline{\underline{x}}_i-\underline{x}_j\right)$$

The overall variability is $ESS = ESS_1 + \cdots + ESS_k$

Is the sum of the distances from the barycentre. The variability within each cluster from its characteristic (barycentre) and is summed.
Note: is NOT multivariate. Is a number that tells how far the clusters are from the barycentre.

The ward algorithm's agglomerate clusters that produce the minimum increase in ESS.
So, among all possible pair of points, you choose the two that, if are in the same cluster, increase less the ESS**.**

We removed the notion of linkage, using only concepts of centres and distances.

Non-hierarchical clustering:

K-means

We need to cluster the training set in k (given) subsets: $C_1, \dots, C_k$
Their union must give X original training set, and their intersection must be empty.

We want to find the group of data such that within each cluster, and summing their variability, I'm minimizing the ESS.
Note: the number of clusters is given. The algorithm won't look for the number of clusters.

Find the clusters that $minimize$:

$$\sum_{i=1}^{k} \sum_{\underline{x}_j \in C_i} d^2\left(\overline{\underline{x}}_i, \underline{x}_j\right)$$

To solve this optimization problem, the solution proposed by k-means:
Initialization step is either 1 or 2:

1- Split the training set randomly into k subsets
2- Assigned at random k $centroids$

Until convergence repeat:

1- Compute centroids $\overline{\underline{x}}_1, \overline{\underline{x}}_2, \dots, \overline{\underline{x}}_k$ such that: $\overline{\underline{x}}_i = \arg\min_{\underline{x}} \sum_{\underline{x}_i \in C} d^2(\underline{x}_i, \underline{x})$
2- Assign each point x to its closest centroid
3- Do the centroids change?
    a. Yes: continue repeating
    b. No: stop

This does not guarantee that the optimization problem is solved: you can converge on a local minimum.
The solution heavily depends on the initialization step: so, run it multiple times to see which clustering reached the minimum cost function.

Difficulties of this algorithm:

- Proving the convergence is not easy
- Computing the centroids, is not easy: **the centroid can be a point that is not in the training data.**

However, the centroid calculation can be simplified, by calculating the medoid (**k-medoids**):

$$\overline{\underline{x}}_i = \arg\min_{\underline{x} \in C} \sum_{\underline{x}_i \in C} d^2(\underline{x}_i, \underline{x})$$

Its main power is that is parameter-less: you only need to specify the distance (and the number of clusters).

A recap of previous algorithms pro and cons

Ward's algorithm, complete linkages, average linkage and k-means tends to form ellipsoidal clusters (by the POV of the distance you give)
One that doesn't have this problem is the **single linkage**. However has the **chain effect:** tends to create clustering structure that mix the **true** clusters.

Density based clustering: DBSCAN

Idea: clusters can be seen as regions with high density. Low density regions can be either bounds between clusters, noise or outliers.
Some points will be not clustered, DBSCAN leaves those points not in a cluster, calling them noise.

Defining a neighbourhood in R^p: for $\varepsilon > 0$ and $\underline{x} \in R^p$, the neighbourhood of $\underline{x}$ is a spherical region (wrt the metric B chosen) with radius ε:

$$N_\varepsilon(\underline{x}) = \{\underline{y} \in R^p : d(\underline{x}, \underline{y}) \leq \varepsilon\}$$

$$|N_\varepsilon(\underline{x}_i)| = number\ of\ points\ in\ X\ that\ are\ neighbours\ of\ \underline{x}_i$$

Two parameters:

- $\varepsilon > 0$
- $\boldsymbol{minpts} \geq 1$**:** minpoints. An integer. Is the minimum number of neighbours that a point must have to be a core point. $\underline{x}_i$ is a **core point** if $|N_\varepsilon(\underline{x}_i)| \geq minpts$. Otherwise, is a **border point.**

- $|N_\varepsilon(\underline{x}_i)| \geq minpts \rightarrow$ $\underline{x}_i$ is a **core point**
- $0 < |N_\varepsilon(\underline{x}_i)| < minpts \rightarrow$ **border point**
- $|N_\varepsilon(\underline{x}_i)| = 0 \rightarrow$ **noise**

We classified the points. How do we build the clusters?
We must think about **dense reachability**: how can we reach points by staying into regions with high density?

$\underline{x}_j$ is **directly density reachable** from $\underline{x}_i$ if $\underline{x}_i$ is a core and $\underline{x}_j$ is in the neighbourhood of $\underline{x}_i$

$\underline{x}_j$ is **density reachable** from $\underline{x}_i$ if you can find a finite sequence of core points $\underline{y}_1, .., \underline{y}_k$ where the first one ($\underline{y}_1$) is $\underline{x}_i$ and the last ($\underline{y}_k$) is $\underline{x}_j$. Moreover, any point of this sequence must stay in the neighbours of the previous: $\underline{y}_j \in N_\varepsilon(\underline{y}_{j-1})$.

$\underline{x}_j$ and $\underline{x}_i$ are **density connected** if there is a point $\underline{x}$ in the training set such that both $\underline{x}_i$ and $\underline{x}_j$ are density reachable from it.

DBSCAN identify the region C such that:

1- If $\underline{x}_j$ is density reachable from $\underline{x}_i$, that belongs to C, then also $\underline{x}_j \in C$
2- Every pair $(\underline{x}_i, \underline{x}_j)$ in C must be density connected

DBSCAN is heavily dependent on the two parameters: fine tune them.

Multidimensional scaling

Having a dataset, and a distance, can we create a different dataset that preserves the chosen distance? We'd like to pass from dataset living in p, to a dataset living in $q \ll p$.
In the new dataset we want to use the Euclidean distance.
The original distance on the original dataset can be any distance: $d(\underline{x}_i, \underline{x}_j) = \delta_{ij}$
We want to create new points y_i, y_j such that:

$$d_{euclidean}\left(\underline{y}_i, \underline{y}_j\right) = \tilde{\delta}_{ij} \cong \delta_{ij}$$

The hard of this is that rigid transformation satisfy this, however, is not interesting. Representation is invariant on those transformations. Hence, we have infinite choice of transformations, but not all of them should be considered.
So, representation is not unique, but who cares? We don't need uniqueness. We only need a representation.

Classical MDS

Find the y-s such that $\min \sum_{i,j}(\tilde{\delta}_{ij} - \delta_{ij})^2$.

Now suppose that also on the original dataset you used Euclidean distance. We want to find the dataset in R^q. We can do this using PCA, by taking only the first $q - components$. Is much more efficient, since we work with covariance matrix that is $p\ x\ p$. In multidimensional scaling we'd use the $n\ x\ n$ matrix.

Kruskal MDS

Find the y-s such that $\min \sum_{i,j}(\vartheta(\delta_{ij}) - \delta_{ij})^2 / \sum \delta_{ij}^2$. Is relative.
This quantity is called stress.
Also with respect to $\vartheta: R \to R$ increasing monotone.

19-4

Regression

We try to explain y (target variable), in terms of $\underline{x}$ (the information we have, the features).

Either to predict y or to explain the variability of y with the features.

The regression function:

$$E[y \mid \underline{x}]: R^p \to R$$

Is the projection of the variable y on the space ($\sigma(\underline{x})$) of all variables on the features are known.

If the join distribution of y and $\underline{x}$ is gaussian ($(y, \underline{x}) \sim N_{p+1}$), we know how to compute the conditional distribution of y given $\underline{x}$, hence we know how the regression function is:

$$E[y \mid \underline{x}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

Is indeed *linear in the regression*.

We have to learn the regression function, since we know only the form, but not the parameters.
An approximation estimated (since we use data) **of the true regressor function** can be **called** $\hat{f}: R^p \to R$. If we believe that the regressor $E[y \mid \underline{x}]$ is sufficiently regular, as first approximation we take a linear representation of the function (like Taylor), but then we must estimate the parameters.
The general model is:

$$y = E[y \mid \underline{x}] + \varepsilon$$

Therefore:

$$y = \hat{f}(\underline{x}) + \varepsilon$$

ε is the residual. Is orthogonal to what we know once we know $\underline{x}$.
ε can't be expressed as a function of $\underline{x}$.

Two approaches:

- No model for $\hat{f}$: **data driven,** let the data speak. The data must be huge, and well built, meaning with lots of sample well representing the population. We'll see **CART.** (NN are a case of data driven regression). Cons: overfit prone, good for prediction, but lack of interpretation.
- Express y with $\underline{x}$ with the most prior knowledge you have. Use this prior knowledge to build **structured models** that explain y. **Linear regression models.** They require a lot of human knowledge, and lots of human effort.

CART (Classification and Regression Trees)

We talk only about regression but is used also for classification.
Split the feature space in subset. For each subset you estimate the y in terms of the features in that subset.

$$Split\ R^p\ into\ R_1, .., R_J\ partitions, and\ predict\ y\ in\ R_i by\ means\ of$$

$$\overline{y}_i = \frac{1}{|R_i|} \sum_{x_j \in R_i} y_j$$

$\hat{f}$ is a step like function:

$$\hat{f}(\underline{x}) = \sum_{j=1}^{J} \overline{y}_i \, 1[\underline{x} \in R_j]$$

Of course, is an approximation of the true f, but is a first approximation, since is a step wise function.

But how do I find the partitions? Their union must reproduce R^p and their intersection is null. But also, I must say that every partition must have some data inside:

$$R_i \cap \{\exists\ k\ x_k\} \neq \emptyset \quad \forall i$$

The optimality criteria to find the partitions (and their number too (J)), is that the MSE from the function step to the true points in each subset, summed, is minimum:

$$min \sum_{j=1}^{J} \sum_{x_i \in R_j} \left(y_i - \overline{y}_j\right)^2$$

Note: if I don't fix J, this will overfit since will have as much as data I have, simply repeating the data.

The optimization problem is NP complete, so CART looks for rectangles $R_1, \dots, R_J$ through a greedy search.

Greedy algorithm to build $\{R_1, \dots, R_J\}$:

1- Look at x_1 and find s_1^* (split) that maximize:

$$\underbrace{\sum_{i=1}^{n} \left(y_i - \overline{y}_j\right)^2}_{\substack{variability\ before \\ the\ split}} - \underbrace{\sum_{x_i < S_i} \left(y_i - \overline{y}_-\right)^2 + \sum_{x_i > S_i} \left(y_i - \overline{y}_+\right)^2}_{\substack{variability\ after \\ the\ split}}$$

   By maximizing this we are reducing the variability.

2- Repeat for $x_2, x_3, \dots, x_p$, therefore you obtain $s_1^*, s_2^*, \dots, s_p^*$ (threshold on each feature)

3- Choose the s_j^* that maximize the cost at step 1 (maximizing the decreasing of variability).

4- Now repeat 1,2,3 for R_1 and R_2 (pretend your dataset is first R_1 and then R_2), so splitting on the subset obtained

5- Stop splitting on R_i if the number of data points inside it is above another threshold (to avoid overfitting)

This procedure can be seen as tree. If you satisfy being above the threshold you split the tree.

The idea is that the role of the features to predict y change based on whether you're, either left of the threshold or right.

Cons:

- Not robust to change in the training set
- Prone to overfit

Pro:

- easy to understand
- **works even if you have missing data:** just stop to a node. Every node is a prediction! If you are close to the root, it means generally lower accuracy prediction, but still, is a prediction
- Same idea used for classification, but you use a different cost function: you count the frequency of labels inside the subset.
- You can works with categorical variables and quantitative at the same time (just different splits)

To control overfitting a strategy is to introduce a penalization term for every unit of the partition. So big partition will be very costly:

$$min \sum_{j=1}^{J} \sum_{x_i \in R_j} \left(y_i - \overline{y}_j\right)^2 + \alpha J$$

α chosen by cross validation: build a huge tree, clearly overfitting the data, and then you start pruning the tree from the bottom, decreasing J and increasing $\sum_{j=1}^{J} \sum_{x_i \in R_j} \left(y_i - \overline{y}_j\right)^2$.

An even better approach is to use ensemble trees: you fix the size of the trees and build several trees.

Linear models for regression

Remark: $\underline{x_i} \in R^p$ is the vector of the observed unit. Our dataset is $X = ((\underline{x}_1, y_1), \dots, (\underline{x}_n, y_n))$

We must design the problem in some engineered way, and for this we use the **Design matrix**:

$$Z = \begin{bmatrix} 1 & z_{11} & \cdots & z_{1r} \\ 1 & z_{21} & \cdots & z_{2r} \\ 1 & \vdots & \cdots & \vdots \\ 1 & z_{n1} & \cdots & z_{nr} \end{bmatrix} \in R^{n\,x\,(r+1)}$$

Where z are transformations of the original dataset, and this transformation is up to you!
There are r functions, problem specific.

$$z_1 = h_1(x_1, \dots, x_p)$$
$$\vdots$$
$$z_r = h_r(x_1, \dots, x_p)$$

The model is linear in the transformed variables z_{ij}, but **NOT LINEAR ON THE ORIGINAL VARIABLES**.

Without assuming Gaussianity, the expected value of the target, given the information we have:

$$E[y \mid \underline{x}] = \beta_0 + \beta_1 h_1(x_1, .., x_p) + \beta_2 h_2(x_1, .., x_p) + \cdots + \beta_p h_p(x_1, .., x_p)$$
$$= \beta_0 + \beta_1 z_1 + \cdots + \beta_p z_p$$

$\beta_0, \beta_1, \dots, \beta_p$ are unknown parameters.

Model for the phenomenon (relationship between the variables), that means for the single response:

$$y = \beta_0 + \beta_1 z_1 + \cdots + \beta_p z_p + \varepsilon$$

With $E[\varepsilon] = 0, Var(\varepsilon) = \sigma^2$. Is explaining the variability of y that is not captured by the variables. For instance, accounts for measurements errors and effect of variables not considered in the model. **Here we're not assuming Gaussianity**. If y and z were gaussian, this would be the exact representation. But in general case, as it is for now, this is an $approximation\ of\ E[y \mid \underline{x}]$.

But we need to express the model for the data (here there will be changes during the course, for instance: will we model also the dependence between units?). The first model will consider all the units independent. But then for linear mixed models and repeated measurements, this will change.
Model for the data (for now: all units are independent):

$$\underline{y} = Z\underline{\beta} + \underline{\varepsilon}$$
$$E[\underline{\varepsilon}] = 0, \qquad Cov(\underline{\varepsilon}) = \sigma^2 I$$

This means that for every unit I have same variance (that is not true for repeated measurements for instance, every unit has its own variability), and also the units are uncorrelated! Since off diagonal terms are null. Is a very hard assumption that we're making.

$$Cov(\underline{\varepsilon}_i, \underline{\varepsilon}_j) = \begin{cases} \sigma^2 & if\ i = j \\ 0 & if\ i \neq j \end{cases}$$

Rephrasing ANOVA as linear model
What we observe in ANOVA are the x_{ij}, hence:

$$y = (x_{11}, \dots, x_{n1}, x_{12}, \dots, x_{n2}, \dots., x_g, \dots, x_{ng}) \in R^n$$

The Z in ANOVA will have second column indicating membership to group 1, the third will indicate membership to group 2, and so on.
The parameter $\underline{\beta}$ is a vector with the baseline and the treatment effects:

$$\underline{\beta} = (\mu, \tau_1, .., \tau_g)$$
$$\underline{\varepsilon} \sim N_n(\underline{0}, \sigma^2 I) \quad \text{since variance is same in each group}$$

The linear model is:

$$\underline{y} = Z\underline{\beta} + \underline{\varepsilon}$$

But rewriting as ANOVA model it ends up being:

$$x_{ij} = \mu + \tau_i + \varepsilon_{ij}$$

The over parametrization of ANOVA in this form reflects also here: the Z matrix has the first column of all ones, that is the sum of all other columns. So the Z matrix is not full rank.

We need full rank if we want a unique solution. But is not a problem.

The constraint used was:

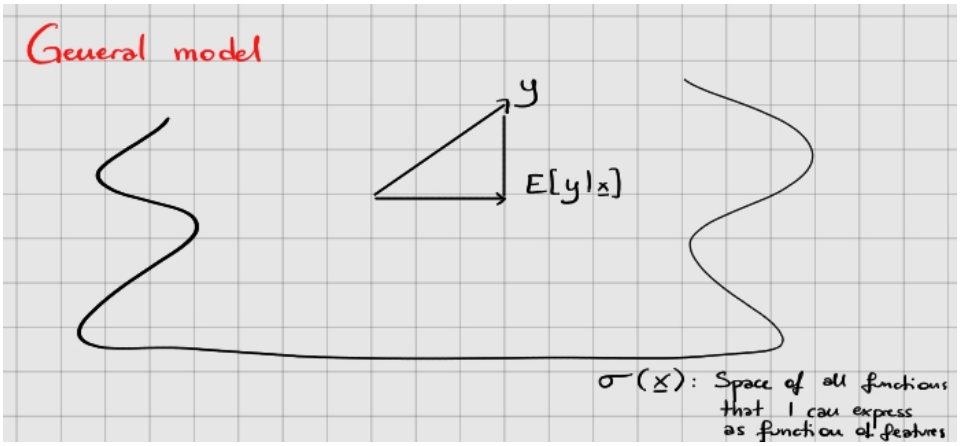
$$\sum_{i=1}^{g} \tau_i n_i = 0 \quad \rightarrow \quad \tau_g = \frac{-\sum_{i=1}^{g-1} \tau_i n_i}{n_g}$$

This will add some terms in the columns after the first, therefore making it full rank, Z is now a $n\ x\ g$ matrix (and not $n\ x\ (g+1)$), and $\underline{\beta} = (\mu, \tau_1, .., \tau_{g-1}) \in R^g$.
Being full rank is not a requirement but makes the analysis easier.

Estimating the parameters $\underline{\beta}$ and σ^2

We need to estimate both the linear coefficients $\underline{\beta}$ and their uncertainty σ^2, this will allow to make tests on the $\underline{\beta}$.

$$\underline{y} = Z\underline{\beta} + \underline{\varepsilon}$$
$$Z\underline{\beta} = E[y \mid Z]$$

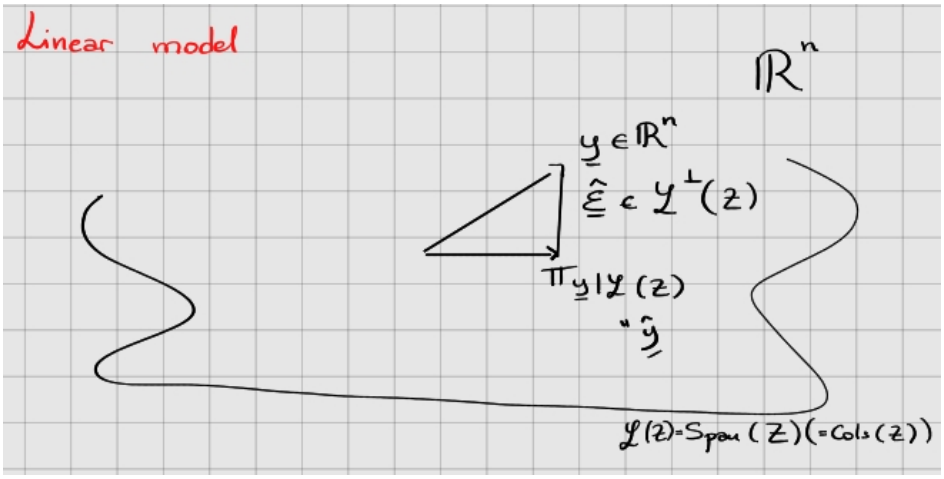


Can we restrict the space to something that we know? Instead of looking in the space of all the possible functions measurable with respect to $\underline{X}$, that is indicated as $\sigma(\underline{X})$, I want to look on a subset of this space: the one generated by the columns of the design matrix Z (that are transformation of $\underline{X}$).

If we see the Z column wise: $Z = [\underline{c}_1, \dots, \underline{c}_{r+1}]$, $\underline{c}_i \in R^n$

$$Z\underline{\beta} = \underline{\beta}_0\underline{c}_1 + \cdots + \underline{\beta}_p\underline{c}_p$$

We are expressing the mean of the target variable as a linear combination of the design matrix: we're looking in the space generated by the columns of the design matrix Z. It's now easier to compute the projection, instead of the general problem.



The projection of the y into the linear space generated by the columns of Z is called fitted value:

$$\hat{\underline{y}} = \pi_{\underline{y}|L(Z)} \quad *$$

Note: $\hat{\underline{\varepsilon}}$ is not the one that appears on the model, it's orthogonal to something you can express as linear function of the Z variables (columns).

From * we can get the $\hat{\underline{\beta}}$, **since is a vector in the space of the columns of Z, must be a linear combination of the z:**

$$\pi_{\underline{y}|L(Z)} = Z\hat{\underline{\beta}}$$

$\hat{\underline{\beta}}$ is the solution of the following optimization problem, since I want him to make the projection on our known linear space (the one generated by the columns of Z) as close as possible to $\underline{y}$:

$$\hat{\underline{\beta}} = \arg\min_{\underline{\beta} \in R^{r+1}} \left|\left|\underline{y} - Z\underline{\beta}\right|\right|^2$$

This is called **OLS: Ordinary Least Square.**
We want to find the coefficient ($\underline{\hat{\beta}}$) of the linear combination of features (transformed features, so the columns of Z) that is the closest to what you observed ($\underline{y}$).
Remember, **we're approximating the regression function, so the expected value of $\underline{y}$ once the regressor have been observed.**
In all possible combinations in $L(Z)$ space (all possible columns of Z), the one that is closer to y.

Proposition: if Z is full rank $(rank(Z) = r + 1 \leq n)$:

1- $\underline{\hat{y}} = \pi_{\underline{y}|L(Z)} = \underbrace{Z(Z^TZ)^{-1}Z^T}_{projection\ on\ L(Z)}\ \underline{y}$

2- Therefore, the solution of the above optimization problem is:

$$\underline{\hat{\beta}} = \underbrace{(Z^TZ)^{-1}Z^T}_{H\ matrix}\underline{y}$$
$$\underline{\hat{\beta}} \in R^{r+1}$$

$H\ matrix$: is the orthogonal projection into $L(Z)$.

23-04

The only thing assumed for now, is that all units are not correlated, that the variance is constant (diagonal term) and the mean of $E(\underline{\varepsilon}) = \underline{0}$.
(Those assumptions will be relaxed in LMM).

Proof of the proposition:
Z^TZ can be expressed with its spectral decomposition.

$$Z^TZ = \sum_{i=1}^{r+1} \lambda_i \underline{e_i}\underline{e_i}^T$$

And since is full rank, the smallest eigenvalue is greater than 0:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{r+1} > 0$$

So,

$$(Z^TZ)^{-1} = \sum_{i=1}^{r+1} \frac{1}{\lambda_i}\underline{e_i}\underline{e_i}^T$$

For every $i = 1, \ldots r+1$, I can define new vectors by transforming the eigenvalues with Z matrix:

$$\underline{q_i} = \sqrt{\frac{1}{\lambda_i}}Z\underline{e_i} \quad \underbrace{\in L(Z)}_{\substack{Since\ is\ a \\ linear\ combination \\ of\ the\ columns\ of\ Z}}$$

$$\underline{q_i}^T\underline{q_j} = \sqrt{\frac{1}{\lambda_i\lambda_j}}\underline{e_i}^TZ^TZ\underline{e_j} = \sqrt{\frac{1}{\lambda_i\lambda_j}}\lambda_j\underline{e_i}^T\underline{e_j} = \begin{cases} 0\ if\ i \neq j \\ 1\ if\ i \neq j \end{cases}$$

Therefore $\left\{\underline{q_1}, \underline{q_2}, \ldots, \underline{q_{r+1}}\right\}$ is an orthonormal basis for $L(Z)$.

$$\pi_{\underline{y}|L(Z)} = \sum_{i=1}^{r+1} \frac{\underline{q_i}\underline{q_i}^T}{\underline{q_i}^T\underline{q_i}}\underline{y} = \sum_{i=1}^{r+1} \frac{1}{\lambda_i}Z\underline{e_i}\underline{e_i}^TZ^T\underline{y} = Z\underbrace{\left(\sum_{i=1}^{r+1} \frac{1}{\lambda_i}\underline{e_i}\underline{e_i}^T\right)}_{(Z^TZ)^{-1}}Z^T\underline{y} = Z(Z^TZ)^{-1}Z^T\underline{y}$$

So we proved that $\hat{y} = Hy$, and $\hat{\beta} = Z(Z^TZ)^{-1}Z^T\underline{y}$.

Case 2: if $rank(Z) = k < r + 1$, so if the determinant of $Z^T Z$ is close to zero. It means that some columns are linearly dependent. Called **collinearity.** We can't do the inverse, we must do the **generalized inverse**, so summing until the last positive eigenvector:

$$(Z^T Z)^- = \sum_{i=1}^{k} \frac{1}{\lambda_i} e_i e_i^T$$

Geometrically is the same thing, you end up only in a smaller space, of dimension k, instead that $r + 1$.
So you have $\hat{\beta} = Z(Z^T Z)^- Z^T \mathrm{y}$. That is **still a linear transformation of the target variable.**

Case 3: if $rank(Z) = r + 1 = n$. We have **number of columns equal to the number of units**. We have in this case that the dimension of $L(Z)$ is n. So we project the y into its same space! So the projection of y into R^n is y. $H = I$, and it means that we're interpolating the data:

$$\hat{y} = y$$

Instead of doing the line in the middle of the points for the typical regression, you have a polynomial interpolating each point. **Complete overfitting.**
Say that we have 5 points on the plane, and in this case we have:

$$Z = \begin{bmatrix} 1 & x_1 & \cdots & x_1^5 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_5 & \ldots & x_5^5 \end{bmatrix}$$

Extremely **high generalization error**, completely overfitting. **Zero bias**, extremely high variance: if you change a point the whole curve change.
You do it only if you don't have variability.
So only if you assume no variability: $Cov(\underline{\varepsilon}) = \underline{0}$.

Let's continue with full rank hypothesis, so we have:

- $\underline{\hat{y}} = H\underline{y}$
- $H = Z(Z^T Z)^{-1} Z^T$ orthogonal projector on $L(Z)$
- $\underline{\hat{\varepsilon}} = \underline{y} - \underline{\hat{y}} = \underbrace{(I - H)}_{orthogonal\ projector\ on\ L^{ortho}(Z)} \underline{y}$

The fitted vector lives in a linear space, and the residual vector lives in its orthogonal space.
The dimensions of the spaces are called **Degrees of Freedom.**

$$L(Z) = span\left(col(Z)\right), \dim(DoF) : r + 1$$
$$L^{ortho}(Z) = \dim(DoF) : n - (r + 1)$$

You have n information, you use $r + 1$ to estimate the model, and you are left with $n - (r + 1)$ degrees to explain the residual, so to estimate σ^2, the meter of the model.
1st Decomposition of variance formula, Pitagora:

$$\left||y|\right|^2 = \left||\hat{y}|\right|^2 + \left||\varepsilon|\right|^2$$
$$SS_{tot} = SS_{reg} + SS_{res}$$

Notice that the vector $\underline{1}$ is in the linear space generated by columns of Z, since it's its first column.

The projection on the space where there is no variability is the sample mean in each component:

$$\pi_{\underline{y}|\underline{1}} = \overline{y} * \underline{1}$$

The projection of the fitted value in that space (what is the mean of the fitted values?):

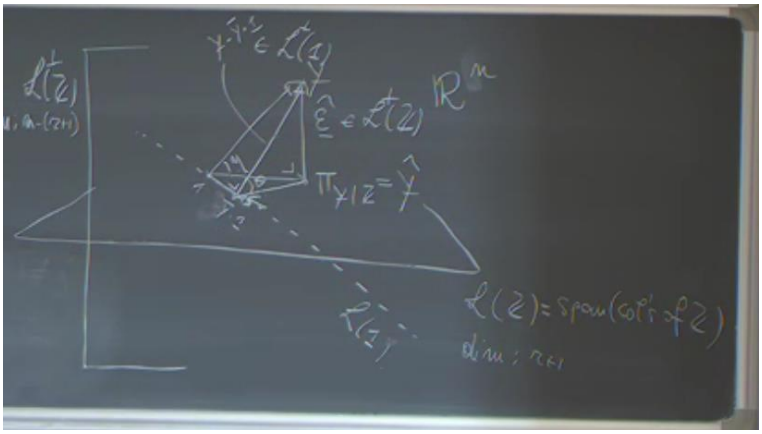
$$\pi_{\underline{\hat{y}}|\underline{1}} = \frac{\underline{1}\underline{1}^T}{\underline{1}^T\underline{1}}\hat{y} = \frac{\underline{1}\underline{1}^T}{\underline{1}^T\underline{1}}Hy$$

And $\underline{1}^T H = \underbrace{\left(H^T\underline{1}\right)^T = \left(H\underline{1}\right)^T}_{\substack{Is\ an\ orthogonal\ projection \\ so\ is\ symmetric}}$

$\left(H\underline{1}\right)^T = \underline{1}^T$ Because I'm projecting on the $L(Z)$, but $\underline{1}$ is already in that space.
So:

$$\pi_{\underline{\hat{y}}|\underline{1}} = \frac{\underline{1}\underline{1}^T}{\underline{1}^T\underline{1}}\underline{y} = \overline{y} * \underline{1}$$

We can therefore draw the other triangle closing on the same point of the other triangle:



We do Pitagora on the other triangle, we get the 2nd formula of variance decomposition:

$$\left|\left|\underline{y} - \overline{y} * \underline{1}\right|\right|^2 = \left|\left|\underline{\hat{y}} - \overline{y} * \underline{1}\right|\right|^2 + \left|\left|\underline{\hat{\varepsilon}}\right|\right|^2$$

$$\sum\left(\underline{y_i} - \overline{y}\right)^2 = \sum(\hat{y}_i - \overline{y})^2 + \sum \underline{\hat{\varepsilon}_i}\text{^}2$$

$$CSS \quad = CSS_{regression} + \quad SS_{res}$$

Dividing everything by the left term:

$$1 = \underbrace{\frac{\sum\left(\underline{\hat{y}_i} - \overline{y}\right)^2}{\sum\left(\underline{y_i} - \overline{y}\right)^2}}_{R^2} + \frac{\sum \underline{\hat{\varepsilon}_i^2}}{\sum\left(\underline{y_i} - \overline{y}\right)^2}$$

Coefficient of determination is the portion of variability of y explained by the regression**.** On the numerator there is the variability of the fitted value around the mean, on the denominator the variability of the target value around the mean

$$R^2 = 1 - \frac{\sum \underline{\hat{\varepsilon}_i^2}}{\sum\left(\underline{y_i} - \overline{y}\right)^2}$$

$$\frac{\sum \underline{\hat{\varepsilon}_i^2}}{\sum\left(\underline{y_i} - \overline{y}\right)^2} = \sin^2\theta$$

Therefore

$$R^2 = 1 - \sin^2\theta = \cos^2\theta$$

R^2 explain how much of the columns of Z are useful to explain the variability of y.

R^2 in this formulation explains how much the regression explain the variability of y proportional to the residual without the regression model.

- $R^2 = 0, \theta = \frac{\pi}{2} \rightarrow \underline{\hat{y}} = \overline{y} * 1$: There is nothing in the regression that talks about $\underline{y}$. The best you can do to explain $\underline{y}$ as a function of the regressors, is to not use the regressors, and use its mean
- $R^2 = 1, \theta = 0 \rightarrow \underline{y} = \underline{\hat{y}}$: Perfect fit, interpolation. From the bold definition of R^2, it means it explain everything, so that you overfitted the data.

All of this is possible for how we've built the design matrix.
If we remove the first column -> **regression through the origin. It means that we don't have 1 in the linear space of the columns of Z.**
If it wasn't for the first column the mean of $\underline{\hat{y}}$ wouldn't necessarily coincide with the mean of $\underline{y}$.

Without the first column, the regression passes through the origin, that means that $\beta_0 = 0$.
Can be done if is a request of the problem.
If not specified in software, that you want regression through the origin, it will compute R^2 as usual, yielding values greater than 1 or lower than 0. So another form must be found in this case, that will be calculated if you specify you want regression through the origin.

$\tilde{R}^2 = 1 - \frac{\left\|\underline{\hat{\varepsilon}}\right\|^2}{\left\|\underline{y}\right\|^2} = \cos^2 \eta$. With this formulation in regression to the origin, we don't have the same interpretation of the previous one! It loses the interpretation of the proportion of variability explained by the model but tells how much y is close to $L(z)$.

To evaluate the fit of the model, we must consider the DoFs. On numerator there is the residual with the fitted model, on denominator the residual without the regression model:

$$R_{adj} = 1 - \frac{\frac{\sum \underline{\hat{\varepsilon}}_i^2}{n-(r+1)}}{\frac{\sum \left(\underline{y}_i - \overline{y}\right)^2}{n-1}}$$

On **denominator** is the **unbiased estimator for the variance of $\underline{y}$ without any regressor.**
On **numerator** is the **unbiased estimator σ^2 when you have $r+1$ regressor**.

So is the ratio of two estimator for σ^2 taking into account the degrees of freedom.
This can't be interpreted as the original R^2, but as **indicator of how good is fitting the data**.
If you use too many DoF for the model, the numerator becomes very big (since $n-(r+1)$=0).
Properties of OLS estimators $\underline{\hat{\beta}}$ and $\underline{\hat{\varepsilon}}$
Note that both of them are linear transformation of y.
Assuming Z to be full rank $r+1$.

1- $E\left[\underline{\hat{\beta}}\right] = \underline{\beta}$, *so* $\underline{\hat{\beta}}$ is an unbiased estimator
2- $Cov\left(\underline{\hat{\beta}}\right) = \sigma^2 (Z^T Z)^{-1}$, depends on how the experiment was conducted.
3- $E\left[\underline{\hat{\varepsilon}}\right] = \underline{0}$, (recall: $E\left[\underline{\varepsilon}\right] = \underline{0}$)
4- $Cov\left(\underline{\hat{\varepsilon}}\right) = \sigma^2 (I - H)$ (recall: $Cov\left[\underline{\varepsilon}\right] = \sigma^2 I$)

5- $E\left[\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}\right] = E\left[\sum_{i=1}^{n} \underline{\hat{\varepsilon}}_i{}^2\right] = \sigma^2\left(n-(r+1)\right)$

$$\textbf{corollary} \rightarrow \hat{\sigma}^2 = E\left[\frac{\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}}{\underbrace{n-(r+1)}_{Total\ DoF}}_{DoF\ used\ to\ explain\ y}\right] = \sigma^2\ unbiased\ estimator\ for\ \sigma^2$$

Proof 1:

Plug the definition of $\underline{\hat{\beta}}$ and use that $E\left[\underline{y}\right] = Z\underline{\beta}$ (since $\underline{\beta}$ is a constant)

Proof 2:

$$Cov\left(\underline{\hat{\beta}}\right) = (Z^TZ)^{-1}Z^T \underbrace{Cov(y)}_{=Cov(\underline{\varepsilon})=\sigma^2 I} Z(Z^TZ)^{-1} = \sigma^2(Z^TZ)^{-1}$$

Note: $\boldsymbol{\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_r}$ **are not uncorrelated**, unless $Z^TZ \cong \alpha I$. So, if you can make the columns of Z orthogonal, you have the properties that the coefficients of the regression are uncorrelated. Change one coefficient change the others.

Plan the experiment (when possible) to have the columns orthogonal. In this way, changing one doesn't change the others.

Is useful, because, say that you have a model, but you want to regulate how a feature affect the target variables. For instance, you want to say that inflation doesn't impact the goleador's price. If you reduce the $\hat{\beta}_i$ for the inflation, also others will change, since are correlated!

Proof 3:

$E\left[\underline{\hat{\varepsilon}}\right] = (I-H)E\left[\underline{y}\right] = (I-H)Z\underline{\beta} = 0$ since is projecting on the orthogonal space of that vector

Proof 4:

$Cov\left(\underline{\hat{\varepsilon}}\right) = (I-H)Cov(y)(I-H)^T = \sigma^2(I-H)$ since its an orthogonal projection, idempotent and symmetric

Note: $(I-H)$ is a $n\ x\ n$ matrix, but is not of rank n. Is projecting a vector of dimension n into a space of dimension $n-(r+1)$. Therefore $\det(I-H) = 0$.

Proof 5

$\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}$ is a number, is the length squared of that vector, and I'm taking the expected value of a number. A number is the same of saying the trace of that number. Trace and expected value are linear operators, can be exchanged. **Trace can invert the factors inside.** $tr\left(\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}\right) = tr\left(\underline{\hat{\varepsilon}}\ \underline{\hat{\varepsilon}}^T\right)$.

I passed from number to matrix, summing on the diagonal the same number!

$$E\left[\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}\right] = tr\ E\left[\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}\right] = E\left[tr\left(\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}\right)\right] = E\left[tr\left(\underline{\hat{\varepsilon}}\ \underline{\hat{\varepsilon}}^T\right)\right]$$

$$\underline{\hat{\varepsilon}} = (I-H)\underline{y} = (I-H)\left(Z\underline{\beta} + \underline{\varepsilon}\right) = (I-H)\underline{\varepsilon}$$

$$E\left[tr(I-H)\underline{\varepsilon}\ \underline{\varepsilon}^T(I-H)^T\right] = tr\left((I-H)\underbrace{E\left[\underline{\varepsilon}\ \underline{\varepsilon}^T\right]}_{Cov(\underline{\varepsilon})}(I-H)\right) = tr\left(\sigma^2(I-H)\right) = \sigma^2\left(n-tr(H)\right)$$

$$tr(H) = tr(Z(Z^TZ)^{-1}Z^T) = tr\left(Z^TZ((Z^TZ)^{-1})\right) = tr\left(I_{(r+1)x(r+1)}\right) = r+1$$

$$= \sigma^2(n-(r+1))$$

Introducing Gaussianity assumption on the residual

$$\underline{\varepsilon} \sim N_n(\underline{0}, \sigma^2 I)$$

Errors are uncorrelated (as before, covariance is an identity, the same of before), but now in gaussian world, they are independent too.

1- The **same estimators** ($\underline{\hat{\beta}}$ and $\hat{\sigma}^2 = \frac{\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}}}{n}$) used before, **are MLE** for $\underline{\beta}$ and σ^2. We're maximizing the probability of observe what we've observed. Notice, now dividing by n and not $n-(r+1)$.

2- $\underline{\hat{\beta}} \sim N_{r+1}(\underline{\beta}, \hat{\sigma}^2 (Z^T Z)^{-1})$

3- $\underline{\hat{\varepsilon}} \sim N_n\left(\underline{0}, \sigma^2 (I-H)\right)$ Estimator of the residual is a gaussian in R^n and is supported on a smaller space $R^{n-(r+1)}$!

4- $\underline{\hat{\varepsilon}}$ is independent from $\underline{\hat{\beta}}$

5- $\underline{\hat{\varepsilon}}^T \underline{\hat{\varepsilon}} \sim \sigma^2 \chi^2(\mathrm{n}-(\mathrm{r}+1))$

Proof 2.3.4

Now we have that $\underline{y} = Z\underline{\beta} + \underline{\varepsilon} \sim N_n(Z\underline{\beta}, \sigma^2 I)$

$$\begin{pmatrix} \underline{\hat{\beta}} \\ \underline{\hat{\varepsilon}} \end{pmatrix} = \begin{pmatrix} (Z^T Z) Z^T \\ I-H \end{pmatrix} \underline{y} \sim N\left(\begin{pmatrix} \underline{\beta} \\ \underline{0} \end{pmatrix}, \sigma^2 \begin{bmatrix} (Z^T Z)^{-1} & \underline{0} \\ \underline{0} & I-H \end{bmatrix}\right)$$

To prove it use that the covariance is $\begin{pmatrix} (Z^T Z) Z^T \\ I-H \end{pmatrix} \sigma^2 I \begin{pmatrix} (Z^T Z) Z^T \\ I-H \end{pmatrix}^T$

That is saying that $\underline{\hat{\varepsilon}}$ and $\underline{\hat{\beta}}$ are independent, since off diagonal term are null.

Proof 5

Say that we don't realize that $(I-H)$ is degenerate (has determinant = 0, see before).

If $\underline{\hat{\varepsilon}} \sim N_n\left(\underline{0}, \sigma^2 (I-H)\right)$ was a proper gaussian, the Mahanalobis distance of a gaussian from its mean:

$$(\underline{\hat{\varepsilon}} - \underline{0}) \left(\frac{I-H}{\sigma^2}\right)^{-1} (\underline{\hat{\varepsilon}} - \underline{0})^T \sim \chi^2(\mathrm{n})$$

But the inverse doesn't exist, since has null determinant. We must use the generalized inverse:

$$(\underline{\hat{\varepsilon}} - \underline{0}) \left(\frac{I-H}{\sigma^2}\right)^{-} (\underline{\hat{\varepsilon}} - \underline{0})^T \sim \chi^2(\mathrm{n}-(\mathrm{r}+1))$$

What matters is the space on which I'm decomposing.

02-05

We continue the narration with the assumptions of gaussian residual:

$$\varepsilon \sim N_n(\underline{0}, \sigma^2 I)$$

We want to test $\hat{\varepsilon}$ and $\hat{\beta}$.

The Mahanalobis distance between $\hat{\beta}$ and β:

$$\frac{1}{\sigma^2}\left(\hat{\beta}-\beta\right)^T(Z^TZ)\left(\hat{\beta}-\beta\right)\sim \chi^2(\mathrm{r}+1)$$

We know that also $\underline{\hat{\varepsilon}}^T\underline{\hat{\varepsilon}} \sim \sigma^2\,\chi^2(\mathrm{n}-(\mathrm{r}+1))$. Those two are independent, so if we take their (normalized) ratio, is a fisher distribution:

$$\frac{\frac{1}{\sigma^2}\left(\hat{\beta}-\beta\right)^T(Z^TZ)\left(\hat{\beta}-\beta\right)}{\frac{r+1}{\underbrace{\frac{\underline{\hat{\varepsilon}}^T\underline{\hat{\varepsilon}}}{n-(r+1)}}_{\hat{S^2}}}} \sim F(r+1,n-(r+1))$$

So we get a pivotal quantity, and the confidence region for β:

$$CR_{1-\alpha}(\beta)=\{nu\ \in R^{r+1}:\left(\hat{\beta}-nu\right)^T(Z^TZ)\left(\hat{\beta}-nu\right)\le(r+1)S^2F_{1-\alpha}(r+1,n-(r+1))\,\}$$

A linear combination of $\hat{\beta}$:

$$a\in R^{r+1}\qquad a^T\hat{\beta}\sim N_1(a^T\beta,\sigma^2a^T(Z^TZ)^{-1}a\,)$$

$$\frac{a^T\left(\hat{\beta}-\beta\right)}{\sigma\sqrt{a^T(Z^TZ)a}}\sim N_1(0,1)$$

So:

$$\frac{\frac{a^T\left(\hat{\beta}-\beta\right)}{\sigma\sqrt{a^T(Z^TZ)a}}}{\sqrt{\frac{\underline{\hat{\varepsilon}}^T\underline{\hat{\varepsilon}}}{n-(r+1)\sigma^2}}}=\frac{a^T\left(\hat{\beta}-\beta\right)}{S\sqrt{a^T(Z^TZ)a}}\sim t(n-(r+1))$$

And the one at the time interval for $a^T\hat{\beta}$:

$$CI_{1-\alpha}(a^T\beta)=\{a^T\hat{\beta}\pm S\sqrt{a^T(Z^TZ)^{-1}a}\;t_{1-\frac{\alpha}{2}}(n-(r+1))\}$$

So if we take the vector a with only a 1 in position i:

$$CI_{1-\alpha}(\beta_i)=\{\hat{\beta}_i\pm S\sqrt{diag_i(Z^TZ)^{-1}}\;t_{1-\frac{\alpha}{2}}(n-(r+1))\}$$

That is what the software gives to you.
However, we're more interested in the global confidence interval for alle the betas, so use Bonferroni instead, correcting α.

All the packages give you also the p-value for this test:

$$H_0\colon\beta_i=0,\qquad reject\ H_0\ at\ level\ \alpha\ if\ \frac{\left|\hat{\beta}_i\right|}{S\sqrt{diag_i(Z^TZ)^{-1}}}>\ t_{1-\frac{\alpha}{2}}(n-(r+1))$$

We want to look on the direction:

$$\max_{a\in R^{r+1}}\frac{\left[a^T\left(\hat{\beta}-\beta\right)\right]^2}{S^2(a^T(Z^TZ)a)}=\frac{1}{S^2}\left(\hat{\beta}-\beta\right)^T(Z^TZ)\left(\hat{\beta}-\beta\right)\sim F\left(r+1,n-(r+1)\right)$$

So the simultaneous confidence interval for linear combination of β:

$$SimCI_{1-\alpha}(a^T\beta)=[a^T\hat{\beta}\pm S\sqrt{a^T(Z^TZ)^{-1}a}\sqrt{F_{1-\alpha}(r+1,n-(r+1))}]$$

$$\frac{\hat{\underline{\varepsilon}}^T\hat{\underline{\varepsilon}}}{\sigma^2} \sim \chi^2(\mathrm{n}-(\mathrm{r}+1))$$

From the proof 5:

$$\frac{(n-(r+1))S^2}{\sigma^2} \sim \chi^2(\mathrm{n}-(\mathrm{r}+1))$$

That is a pivotal quantity for σ^2, obtaining the confidence interval:

$$CI_{1-\alpha}(\sigma^2) = [\frac{n-(r+1)S^2}{\chi_{1-\alpha}{}^2(\mathrm{n}-(\mathrm{r}+1))}]$$

If we can't reject that $\beta_i = 0$, we can remove that regressor, simplifying our model.

We want to test p linear combinations of beta, so we build a matrix $C \in R^{p\,x\,(r+1)}$

$$H_0: C\beta = 0$$

$$C\hat{\beta} \sim N_p(C\beta, C\hat{\sigma}^2(Z^TZ)^{-1}C^T)$$

Under H_0:

$$\frac{\frac{\left(C\hat{\beta}\right)^T(C(Z^TZ)^{-1}C^T)^{-1}C\hat{\beta}}{p}}{\frac{\hat{\underline{\varepsilon}}^T\hat{\underline{\varepsilon}}}{\sigma^2(n-(r+1))}} \sim F(p, n-(r+1))$$

The statistic is $F = \frac{1}{S^2}\left(C\hat{\beta}\right)^T(C(Z^TZ)^{-1}C^T)C\hat{\beta} \sim pF(p, n-(r+1))$
So we reject if $F > pF_{1-\alpha}(p, n-(r+1))$

We always aim to remove some regressor beta from the model in order to better explain the variability (our meter of explanation):

$$H_0: \beta_r = \beta_{r-1} = \cdots = \beta_{r-(p-1)} = 0$$

$$C = \begin{bmatrix} 0 & 0 & \dots & 1 & \square & \square \\ 0 & 0 & & \square & 1 & \square \\ \vdots & & \ddots & & \vdots & \\ 0 & 0 & \dots & \square & \square & 1 \\ 0 & 0 & & \underbrace{\square \quad \square \quad \square \quad 1}_{p} & & \end{bmatrix} = [0\ I_p] \in R^{p\ x\ (r+1)}$$

We can organize the POV as:

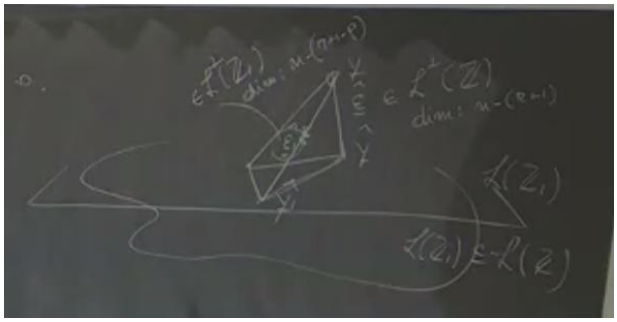
$$Z = [Z_1 \quad \underbrace{Z_2}_{The\ regressor\ we\ want\ to\ see\ if\ are\ 0}]$$

The testing can be seen as:

$$Full\ model: y = Z\beta + \varepsilon$$
$$Reduced\ model: y = Z_1\beta_1 + \varepsilon$$

Remember: Z_1 lives in the dimension $r + 1 - p$
Z_2 lives in the dimension p
Z lives in the dimension $r + 1$

If the last regressor are zero, the two $\hat{y}$ ($\hat{y}\ and\ \hat{y}_1$) are not really different. So we're wasting many DoF for not really much.



If we reject the hypothesis, it means that we have to use the full model.

$SS_{res}(Z_1)$: length of the ipotenusa of the oblique triangle (the one with $\hat{y}_1$)
$SS_{res}(Z)$: length of the ipotenusa of the starting triangle (the one with $\hat{y}$)

The test can be seen as, reject H_0 if $SS_{res}(Z_1) - SS_{res}(Z)$ is large:

$$\frac{SS_{res}(Z_1) - SS_{res}(Z)}{S^2 p} \sim F(p, n - (r + 1))$$

If, in particular, all the $r + 1$ betas are zero, it means that the y can't be explained at all by the features, and $SS_{res}(Z_1) = \sum(y_i - \bar{y})^2$.
So, without using regressor, the best thing to explain y is the mean.
The test becomes (and is what R does in the **F test**):

$$\frac{SS_{res}(Z_1) - SS_{res}(Z)}{S^2 r} = \frac{\sum(y_i - \bar{y})^2 - \sum \hat{\varepsilon}_i^2}{S^2 r} = \frac{\frac{\sum(y_i - \bar{y})^2}{r}}{\frac{\sum \hat{\varepsilon}_i^2}{n-(r+1)}} \sim F\big(r, n-(r+1)\big)$$

Rejecting if there is at least one regressor useful to explain y.
Reject if $F > F_{1-\alpha}(r, n-(r+1))$

If we can't reject this, we can't run a linear model that is useful in any scenario.

Prediction

We have a new datum $z_0 = (1 \;\; z_{0_1} \; z_{0_2} \ldots z_{0_r})$ not in the training set.

We predict as: $E[y_0|z_0] = \; z_0^T\beta$, and is unbiased prediction of $z_0^T\hat{\beta} = z_0^T(Z^TZ)^{-1}Z^Ty$

Gauss Markov theorem says that $z_0^T\hat{\beta}$ is BLUE: Best Linear Unbiased Estimator of $z_0^T\beta$

The fitted value is not giving the y_0, but the expected value of y_0.

The confidence interval for the prediction is the same of before, but replacing a with z_0:

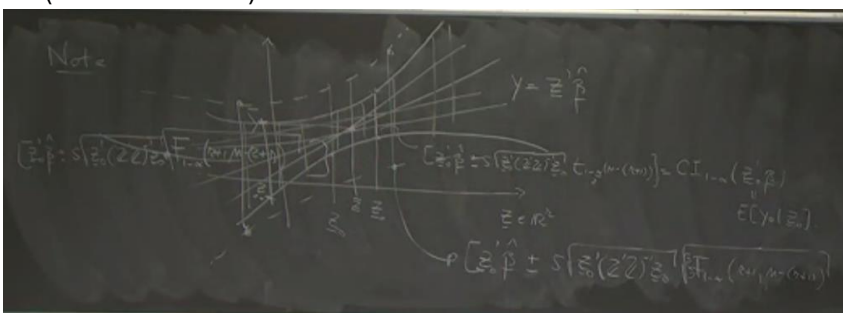
$$CI_{1-\alpha}({z_0}^T\beta) = \{{z_0}^T\hat{\beta} \pm S\sqrt{{z_0}^T(Z^TZ)^{-1}z_0}\; t_{1-\frac{\alpha}{2}}(n-(r+1))\}$$

If we take several different values (z_0) to predict we get several t intervals (the one on the y axis). Practitioners would say that the true model is in the band generated by all those intervals with confidence $1-\alpha$, but is wrong! Those are one at the time confidence intervals.

The true model lives in the band generated by those simultaneous intervals:

$$SimCI_{1-\alpha}(z_0^T\beta) = [z_0^T\hat{\beta} \pm S\sqrt{{z_0}^T(Z^TZ)^{-1}z_0}\sqrt{F_{1-\alpha}(r+1, n-(r+1))}]$$

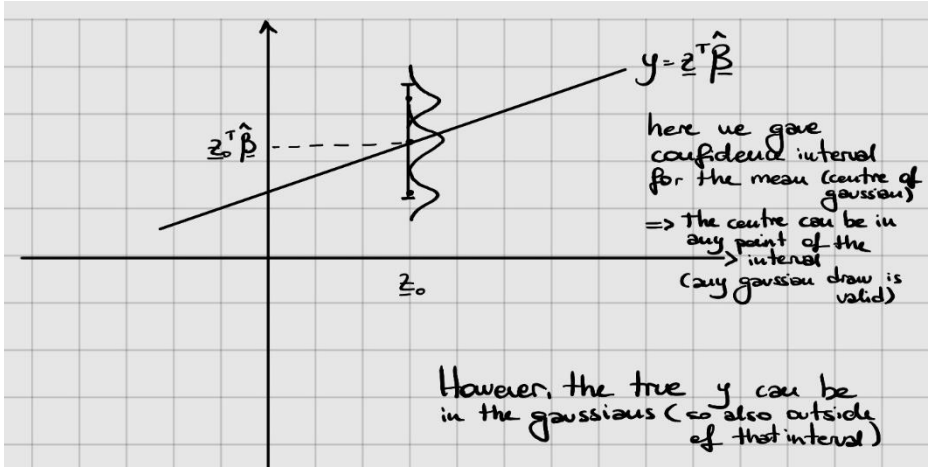
Obviously, this band (the dashed one) will be wider that the one at the time.



There is variability in our estimator $z_0^T\beta$, so the true y will be around this point $z_0^T\hat{\beta}$ (the mean).
There is uncertainty ε_0 for him, since for the true model $y_0 = z_0^T\beta + \; \varepsilon_0$.
y_0 lives somewhere (in the gaussian form) near $z_0^T\beta$. When we estimate the mean, we estimate with $z_0^T\hat{\beta}$, but here we have variability.
What we've done so far, is the confidence interval for the $z_0^T\beta$.
In other words, we have two uncertainties:

- One due to the fact that y_0 is distributed around its mean
- One on where the mean is (we've estimated it giving the confidence interval, but we don't know if is the true mean). This uncertainty is what was faced before, with the previous confidence interval.

We now want an interval I such that contains the true y_0 once I know the regressor, with probability $1-\alpha$: $P[y_0 \in I \,|z_0] = 1-\alpha$.
Here we are taking into account also the residual ε_0.



Confidence interval for the prediction:
We're not looking for the mean of y_0 ($E[\widehat{y_0}\,|z_0]$) to be in this interval, but y_0 himself!

$$y_0 \sim N(z_0^T\beta, \sigma^2)$$
$$z_0^T\hat{\beta} \sim N({z_0}^T\beta, \sigma^2(Z^TZ)^{-1})$$

Those two are independent, since ε_0 is independent from $\varepsilon_1, \dots, \varepsilon_n$ because is a new observation, not from the training set, so:

$$y_0 - z_0^T\hat{\beta} \sim N(0, \sigma^2(1 + z_0^T(Z^TZ)^{-1}z_0)$$

Therefore, this ratio is a ratio of two normal distributions (denominator is chi squared divided by its DoF) and are independent since the denominator is the residual of the model:

$$\frac{\dfrac{y_0 - z_0^T\hat{\beta}}{\sqrt{\sigma^2(1 + z_0^T(Z^TZ)^{-1}z_0)}}}{\sqrt{\dfrac{1}{\sigma^2}\dfrac{\hat{\varepsilon}^T\hat{\varepsilon}}{n-(r+1)}}} \sim t(n-(r+1))$$

➔ Our pivotal

The interval is:

$$P[y_0 \in \left[z_0^T\hat{\beta} \pm S\sqrt{1 + z_0^T(Z^TZ)^{-1}z_0}t_{1-\alpha}(n-(r+1)\right]] = 1-\alpha$$

Is like the interval for the mean, with a +1 in more: this +1 is because there is extra variability generated by ε_0, that is another σ^2. Obviously is a larger interval.

3-5

So far we've assumed the error terms having same variance, and null covariance.
What if now, we go more general:

$$Cov(\varepsilon) = \sigma^2\Sigma$$

Assuming Σ is known, and σ^2 is unknown, so we only know the structure of the covariability apart from a multiplicative term.

Now we can't estimate the betas as before, since we've used Euclidean distance. Now we must instead consider the Mahanalobis distance to the target y:

$$\hat{\beta} = \arg\min_{\beta}(y - Z\beta)^T\Sigma^{-1}(y - Z\beta)$$

This is called **Generalized Lest Squares.**

This minimization problem is easy if we know Σ, indeed:

$$(y - Z\beta)^T\Sigma^{-1/2}\Sigma^{-1/2}(y - Z\beta) = \underbrace{\left(\underbrace{\Sigma^{-1/2}y}_{\tilde{y}} - \underbrace{\Sigma^{-1/2}Z}_{\tilde{Z}}\beta\right)^T\left(\Sigma^{-1/2}y - \Sigma^{-1/2}Z\beta\right)}_{||\tilde{y} - \tilde{Z}\beta||}$$

So

$$\hat{\beta} = \arg\min_{\beta}(y - Z\beta)^T\Sigma^{-1}(y - Z\beta) = \arg\min_{\beta}||\tilde{y} - \tilde{Z}\beta||$$

$\hat{\beta} = \cdots\ldots = (Z^T\Sigma^{-1}Z)^{-1}Z^T\Sigma^{-1}y$

We're back to the OLS with new transformed variables. Notice that the transformed variables are such that the covariance is the same of OLS, so $\sigma^2 I$. Here axis is uncorrelated with same variance. This is done by multiplying $\Sigma^{-1/2}$:

$$y = Z\beta + \varepsilon$$
$$\Sigma^{-1/2}y = \Sigma^{-1/2}Z\beta + \Sigma^{-1}\varepsilon$$

If Σ is unknown, you have some solutions:

- Parametrize Σ, but with this solution you can't use the trick of before, since you must model Σ in some way, so you must assume Gaussianity and use ML estimator
- Estimate Σ directly from the data iteratively: start assuming is the identity (so go with OLS), then look at the residual, then use the obtained Σ look at residual and so on
- Transform the regressor Z (or the target variable y), or both.

Situations when you know Σ

When target variable y_i is the mean of n_i observations independent with some variability σ^2.
In this case $Var(y_i) = \frac{\sigma^2}{n_i}$, so:

$$\Sigma = \begin{bmatrix} 1/n_1 & \square & \square & \square \\ \square & 1/n_2 & \square & \square \\ \square & \square & \dots & \square \\ \square & \square & \square & 1/n_n \end{bmatrix}$$

We're **weighting units**: units with higher n_i have higher weights (since you consider Σ^{-1}), because you are less uncertain (because you have bigger sample for them).

$$\Sigma^{-1} = \begin{bmatrix} n_1 & \square & \square & \square \\ \square & n_2 & \square & \square \\ \square & \square & \dots & \square \\ \square & \square & \square & n_n \end{bmatrix}$$

For this reason, called **WLS.**

When target variable y_i is the sum of n_i observations independent with some variability σ^2.
In this case $Var(y_i) = \sigma^2 n_i$, so is the reciprocal of before:

$$\Sigma^{-1} = \begin{bmatrix} 1/n_1 & \square & \square & \square \\ \square & 1/n_2 & \square & \square \\ \square & \square & \dots & \square \\ \square & \square & \square & 1/n_n \end{bmatrix}$$

Is **always WLS**, with different weights!

Diagnostic of Linear Models

Look for:

- outliers, heteroschedasticity, normality, autocorrelation on the residuals.
- influential cases in the units
- collinearity
- many more that depends on the specific problem

Residual analysis (of $\hat{\varepsilon}$)

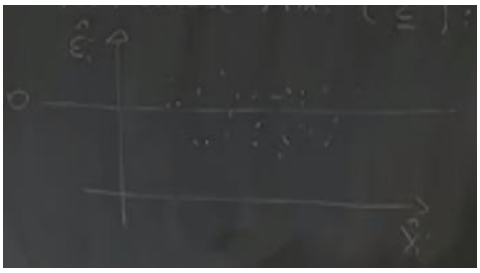
Remark:

- Abstract model: $y = Z\beta + \varepsilon$
  - $E[\varepsilon] = 0$ and $Cov(\varepsilon) = \sigma^2 I$
  - $\varepsilon \sim N(0, \sigma^2 I)$ **distribution in $\boldsymbol{R^n}$**

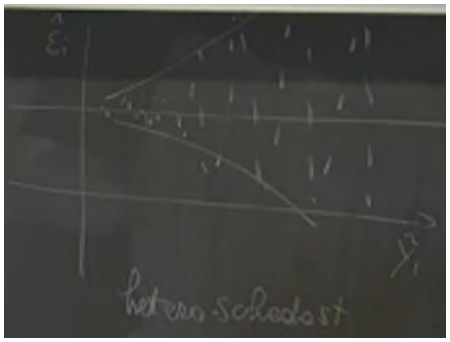
- Fitted model: $y = Z\hat{\beta} + \hat{\varepsilon}$
  - $E[\hat{\varepsilon}] = 0$ and $Cov(\hat{\varepsilon}) = \sigma^2(I - H)$
  - With normality assumption of residual: $\hat{\varepsilon} \sim N(0, \sigma^2(I - H))$ **distribution in $\boldsymbol{L^{orthog}(Z)}$**

We can see how the estimated residual $\hat{\varepsilon}$ lives on a different space than the true one. We use it since is the only information we have, and we use it for criticize the abstract model.

Cloud of point around 0 since has zero mean. If we see something like this, we'd be happy:

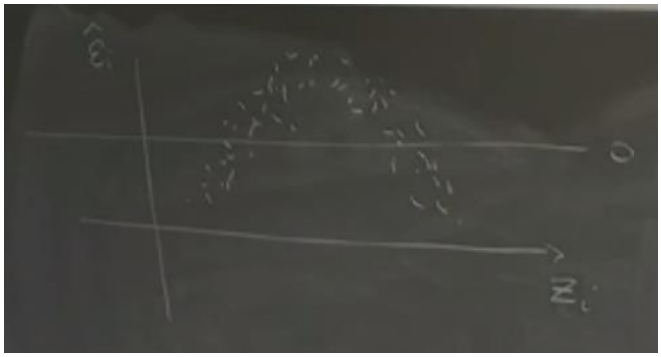


Typically, we have that the larger the fitted value, the larger the error. So, the variability is changing with y. This is against our assumptions that the variance is the same for every unit. Is called indeed **heteroschedasticity**:



Transform the variable to reduce this: since you use a simple model (bias variance trade-off/Occam razor).

By plotting residual against a single regressor, you may see this:



The residual is dependent on the regressor: in a region is positive and in the rest is negative.
But they should be orthogonal!
Solution: transform z_i. How? Depends, as always. A solution in this case is with z_i^2:

In this way you can't explain the error with the regressor.
Remark: any transformation is plausible, trigonometric functions (since capture periodicity).

$Var(\hat{\varepsilon}_i) = \sigma^2(1 - h_{ii})$, where $h_{ii} = diag(H) = diag(Z(Z^TZ)^{-1}Z^T)$
Rather than analysing $\hat{\varepsilon}_i$, look at the **studentized residual** to capture outliers of residual:

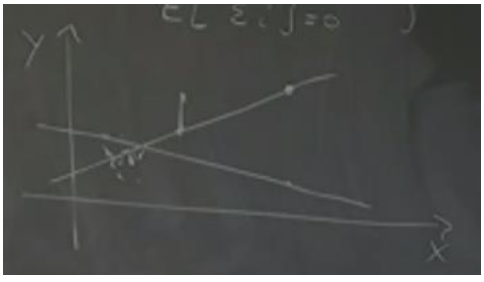
$$\hat{\varepsilon}_{i_{studentized}} = \frac{\hat{\varepsilon}_i}{S\sqrt{1 - h_{ii}}}$$

With this, residual components have more similar variance. In this way outliers of regressor are removed (and is important since we're working in L_2).
If outliers disappear after transforming it with studentized residual, then the problem is in the design of the problem (since depends on h_{ii}, that himself depends on Z).

h_{ii} are called **leverages** since they increase the regression line (a single point screw up the entire model): $h_{ii} \in [0,1]$, because of idempotency of H:

- If $h_{ii} = 1 \rightarrow Var(\hat{\varepsilon}_i) = 0$.

Leverages are due to bad problem design: it doesn't depend on the target variable y or data.
Leverage effect:



Influential cases

Let fix notation: $Z_{-i} = Z\ without\ i-th\ row$.
Same for target variable y_{-i}.

We can fit the model without a unit:

$$y_{-1} = Z_{-1}\beta_{-1} + \varepsilon_{-1}$$

From here we get $\hat{\beta}_{-1}$.

If $\hat{\beta}_{-1}$ and $\hat{\beta}$ are very different, it means that the $i-th$ unit is influential: you get a completely different model if you consider that unit or not.
Different with respect to **Cook distance** that is the Mahanalobis distance, divided by our meter, the variability S^2:

$$D_i = \frac{(\hat{\beta} - \hat{\beta}_{-1})(Z^T Z)(\hat{\beta} - \hat{\beta}_{-1})}{S^2(r+1)}$$

If D_i is big, the $i-th$ unit is influential.
Compare D_i with the $F(r+1, n-(r+1))$ quantile, since if you fix $\hat{\beta}_{-1}$ D_i has an F distribution.
Software prints out D_i only if is greater than 1, as a rule of thumb.

D_i can also be written as:

$$D_i = \left(\frac{\hat{\varepsilon}_i}{S\sqrt{1-h_{ii}}}\right)^2 \frac{h_{ii}}{1-h_{ii}} \frac{1}{r+1}$$

$\boldsymbol{D_i}$ can be **large either if there is an high leverage** (the second component) **or if you have high studentized residual** (or both).
So **influential cases can be outliers, influential cases, or both**. Or a combination of the two (that may be acceptable), when combined create problem

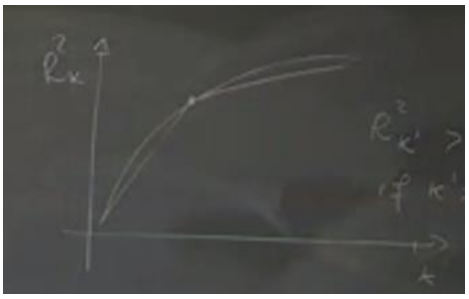
Leverage points are identified based on the predictor variables, while outliers are identified based on the response variable.
A leverage point can be an outlier if it also has an unusual value for the response variable, but this is not always the case.
Conversely, an outlier in the response variable does not necessarily have high leverage.

Model selection

What are the variables that should enter the model or not?
Given r variables (so r regressors), I can build up to 2^r models since each variable can either enter or not.
Is hard computationally to explore all the possible models but is possible for small r.
$For\ k\ in\ 1,..,r$:

- $Fit\ all\ \binom{r}{k}\ models$
- $Chose\ the\ best\ one\ (i.e.\ the\ one\ with\ highest\ R^2)$
- $Store\ R_k^2\ in\ a\ list$

$Plot\ the\ list\ and\ chose\ based\ on\ elbow\ method$:



Alternatives are to use other performance indexes instead of R^2.
At the end I'll fit all the r models, for small r.

You can go more heuristically, not exploring all the possible models.

Lasso and Ridge will face both collinearity and model selection.

6-5

Collinearity and variable selection

Are 2 problems quietly related.
Z collect observations of the covariates of r regressors on n statistical units.

Collinearity: two or more regressors (the columns of Z) are *close* to be linear dependent.
Although we have r regressor, we have something that can be explained as a linear combination of 2 or more regressors.
This is a problem, because with OLS we estimated the $\hat{\beta}$ (if Z is full rank) as $\hat{\beta} = (Z^T Z)^{-1} Z^T y$

But if they are linear dependent, the space where I'm projecting is not $r+1$, but smaller. Also, inverting Z^TZ is hard, because is close to be singular.
Moreover, the variability/covariability of $\hat{\beta}$ will explode: $Var(\hat{\beta}) = \sigma^2(Z^TZ)^{-1}$
Hence, we're uncertain of the predictions of our model (since we're unsure on the $\hat{\beta} - s$).

We define the **coefficient of determination** R_j^2 as the R^2 of the model with the z_j as target variable of the model, and as regressor you use the ones of the previous model, without him: $z_1, \dots, z_r$ without the $j-th$.

In this way, we can express in a clearer way the variability of the single $\hat{\beta}_j$:

$$Var(\hat{\beta}_j) = \frac{\sigma^2}{\sum_{i=1}^{n}(Z_{ij} - \overline{Z}_j)^2}\frac{1}{1-R_j^2} = diag_j(\sigma^2(Z^TZ)^{-1})$$

If $\sum_{i=1}^{n}(Z_{ij} - \overline{Z}_j)^2$ increases $\rightarrow Var(\hat{\beta}_j)$ decreases. So is a good idea to have the regressors spread out. Indeed, if you take points all close to each other, you are uncertain on how you should draw the line. This can be done only through the design of experiment.

If R_j^2 increases up to 1 $\rightarrow \boldsymbol{Var(\hat{\beta}_j)}$ **increases**. This is the problem of collinearity: the **variable *j* can be expressed as linear combination of the other regressors.**

Rule of thumb: when $VIF = \frac{1}{1-R_j^2} > 5 \rightarrow collinearity$. Is called Variance Inflation Factor.
5 because $R_j^2 = 0.8$ so the variable j would be explained well at 80% by linear combination of other regressors.
The dream situation is when regressors are orthogonal, so changing a regressor won't impact the other. In this way by looking at the coefficients, we know the impact of that feature on the target.
But $\frac{1}{1-R_j^2}$ is inflating the variability of that regressor, only because of the dependence with other regressors.

What can we do about it? Removing is not a good idea: if a variable is correlated with another, also the other is correlated with the others.
We must cure the problem in some smarter way, otherwise we would remove everything.

We want to directly restrict the variance of the estimator, with **regularization methods**.

To remark, $y_0 = z_0^T\hat{\beta}$.
By giving as z_0 the mean of the columns of the design matrix $z_0 = (1\ \overline{z}_1\ \overline{z}_2 \dots \overline{z}_r) = \frac{Z^T1}{1^T1}$ we get the fitted value $y_0 = \frac{1^TZ}{11^T}(Z^TZ)^{-1}Z^Ty = \frac{1^THy}{1^T1} = \frac{1^T\hat{y}}{1^T1} = \overline{y}$ (remark $(1^TH) = (H1)^T = 1^T$)

Hence, the model goes into the barycentre of the data cloud.

$$\overline{y} = \hat{\beta}_0 + \hat{\beta}_1\overline{z}_1 + \dots + \hat{\beta}_r\overline{z}_r$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}_1\overline{z}_1 - \cdots - \hat{\beta}_r\overline{z}_r$$

$$y_0 - \overline{y} = \hat{\beta}_1(z_{01} - \overline{z}_1) + \cdots + \hat{\beta}_r(z_{0r} - \overline{z}_r)$$

$\hat{\beta}_0$ disappears, since is what allow us to pass through the barycentre. This is the first column of Z were all 1-s. If it wasn't, it was already passing through the origin.
We want to keep the OLS fitted model to pass through the barycentre.
So we always centre the target variable: $y = y^* = y - \overline{y}\,1 = \begin{pmatrix} y_1 - \overline{y} \\ \dots \\ y_n - \overline{y} \end{pmatrix}$.

Centre the design matrix:

$$Z = Z^* = \begin{bmatrix} z_{11} - \overline{z}_1 & z_{12} - \overline{z}_2 & \cdots & z_{1r} - \overline{z}_r \\ z_{21} - \overline{z}_1 & \cdots & \cdots & z_{2r} - \overline{z}_2 \\ \cdots & \cdots & \cdots & \cdots \\ z_{r1} - \overline{z}_1 & \cdots & \cdots & z_{rr} - \overline{z}_r \end{bmatrix}$$

One column is lost, because the first would be 0, so Z has now r columns.

So the OLS becomes:

$$\underline{\hat{\beta}}^* = \arg\min_{\underline{\beta} \in R^r} \left|\left| \underline{y}^* - Z^*\underline{\beta} \right|\right|^2$$

Then go back to the original dataset:

$$\underline{\hat{\beta}}_0 = \overline{y} - \hat{\beta}_1\overline{z}_1 - \cdots - \hat{\beta}_r\overline{z}_r$$

From now on we always centre the variables.

Ridge regression

After centring the variables, we now want to constraint the regressors to be small (inside the circle of radius S^2). Working on restricting the $\underline{\beta}$ we have the effect of reduced variance.

$$\begin{cases} \arg\min_{\underline{\beta} \in R^r} \left|\left| \underline{y} - Z\underline{\beta} \right|\right|^2 \\ \left|\left| \underline{\beta} \right|\right|^2 \leq S \end{cases}$$

We write $\underline{\hat{y}} = H\underline{y} = Z\underbrace{(Z^TZ)^{-1}Z\underline{y}}_{\underline{\hat{\beta}}_{OLS}} = Z\underline{\hat{\beta}}_{OLS}$

The objective function can be rewritten by summing and subtracting $\underline{\hat{y}}$:

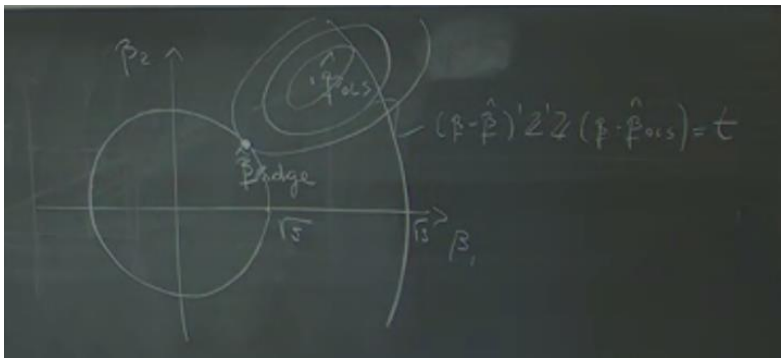
$$\left|\left|\underline{y} - Z\underline{\beta}\right|\right|^2 = \left|\left|\underline{y} - \underline{\hat{y}} - Z\left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)\right|\right|^2$$

$$\left|\left|\underbrace{\underline{y} - \underline{\hat{y}}}_{\underline{\hat{\varepsilon}} \in L^{orth}(Z)} - \underbrace{Z\left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)}_{\in L(Z)}\right|\right|^2 \rightarrow can\ apply\ pitagora = \left|\left|\underline{\hat{\varepsilon}}\right|\right|^2 + \left|\left|Z\left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)\right|\right|^2$$

Therefore the original problem can be written as (since the residual is orthogonal to the space R^r on which I'm looking for the solution) $\arg\min_{\underline{\beta} \in R^r} \left|\left|Z\left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)\right|\right|^2$, that is equivalent to:

$$\arg\min_{\underline{\beta}} \left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)^T Z^T Z \left(\underline{\beta} - \underline{\hat{\beta}}_{OLS}\right)$$

That is an ellipse. Its solution would be the centre ($\underline{\hat{\beta}}_{OLS}$). But we're restricting the solution to be in the circle:



The best solution is, so, the point where is satisfied the constraint for the β to be in the circle.
As $S \rightarrow 0$ I have a sequence of solutions going to zero.

The solution of a constrained optimization problem is found with the Lagrangian:

$$\arg\min_{\underline{\beta} \in R^r} \left|\left|\underline{y} - Z\underline{\beta}\right|\right|^2 + \lambda \left|\left|\underline{\beta}\right|\right|^2$$

Where the penalization λ (Lagrange multiplier) depends on the S.

You want to minimize the $\left|\left|\underline{y} - Z\underline{\beta}\right|\right|^2$ but pay λ for every unit of distance of $\left|\left|\underline{\beta}\right|\right|^2$ from zero

I'm penalizing solutions that are irregular. The higher the λ, the more you penalize big values of $\left|\left|\underline{\beta}\right|\right|$.

Hence, the more you're shrinking the $\underline{\beta}$.

With Lagrangian we can now take derivatives to optimize, obtaining:

$$\underline{\hat{\beta}}_{ridge} = (Z^T Z + \lambda \mathrm{I})^{-1} Z^T y$$

Note: without penalization you go back to standard OLS.

$Z^T Z + \lambda \mathrm{I}$ is now easier to invert, if you have collinearity!

Observations:

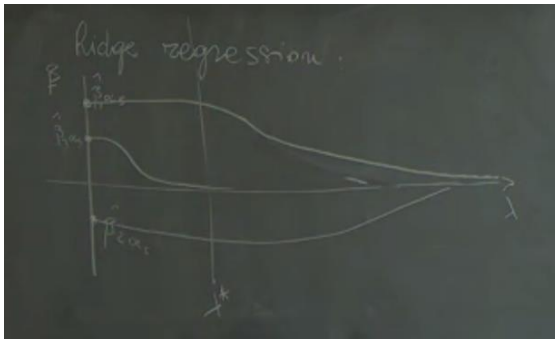
- $\underline{\hat{\beta}}_{ridge}$ estimator is **biased**
- For any regression problem, there is a λ^* such that the mean square error making on the estimator (even if biased), has smaller bias than the mean square error of OLS:

$$E\left[\left|\left|\underline{\hat{\beta}}_{ridge} - \underline{\beta}\right|\right|^2\right] \leq E\left[\left|\left|\underline{\hat{\beta}}_{OLS} - \underline{\beta}\right|\right|^2\right]$$

So, yes, **is biased, but you're making smaller errors (smaller variability**) (closer to the centre of the car park, more far from trees!)

- However, we only know exists some λ^*, but how do you find it? we can't use the first above formula because we don't know $\underline{\beta}$. So, use cross-validation.

The more you penalize the more $\underline{\beta}$ disappears:



In this picture, we see that with that λ^*, β_3 can be removed

Another solution to face collinearity: **PCA regression.** Collinearity doesn't happen if we have orthogonal regressors. Orthogonality is the opposite of collinearity. So perform PCA on Z on column wise. Replace the $\underline{z}_{11}$ with $\underline{s}_{11}$ being the score of unit i of PC_j. You could even perform dimensionality reduction on regressors. Getting fewer column of Z.
The fitted model is (since the columns of Z are now the principal components):

$$\hat{y} = \underline{\hat{\beta}}_{PCA_1} PC_1 + \cdots + \underline{\hat{\beta}}_{PCA_k} PC_k$$

Where $PC_1 = e_{11}z_1 + e_{21}z_2 + \cdots + e_{r1}z_r$
So the fitted model can be rewritten as:

$$\hat{y} = z_1 \underbrace{\left(\underline{\hat{\beta}}_1 e_{11} + \underline{\hat{\beta}}_2 e_{12} + \cdots \underline{\hat{\beta}}_r e_{1r} \right)}_{\hat{\gamma}_1} +$$

$$z_2 \underbrace{\left(\underline{\hat{\beta}}_1 e_{21} + \underline{\hat{\beta}}_2 e_{22} + \cdots \underline{\hat{\beta}}_r e_{2r} \right)}_{\hat{\gamma}_2} + \cdots$$

Obtaining a model that is a linear combination of the original regressors, but the $\underline{\hat{\beta}}_{-s}$ are not the OLS ones.

Problems of both PCA regression and ridge regression doesn't give sparse solutions in terms of $z_1 \dots z_r$. For PCA you'd need some loading $e_{ij} = 0$ to select the variables, but in general case, it doesn't happen. One may choose a sparse space that is given by PCA's solution (that is may different from the one generated by principal components) to automatically select variables. (This is called VARIMAX).

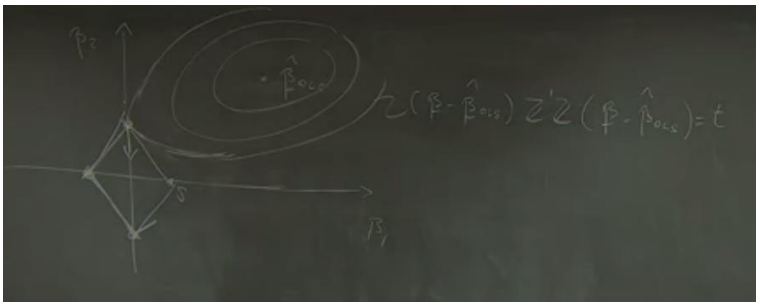
Back to ridge, we'd like to end up in some corner (the dots, where I'd have $\beta_1 = 0 \; and \; \beta_2 = S$ in the bottom picture). By having spikes rather than circles, we have higher chance to be on the corner:

$$\begin{cases} \arg\min_{\underline{\beta} \in R^r} \left|\left| \underline{y} - Z\underline{\beta} \right|\right|^2 \\ \left|\left| \underline{\beta} \right|\right|_1 < S \qquad \sum |\underline{\beta_i}| < s \end{cases}$$

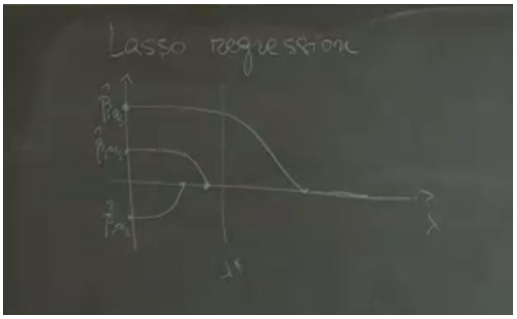
This is called Lasso Regression. Working on the $L - 1$ norm.

The problem is that there is not an analytical solution for this $\arg\min_{\underline{\beta} \in R^r} \left|\left| \underline{y} - Z\underline{\beta} \right|\right|^2 + \lambda \sum |\underline{\beta_i}|$

Here you're both reducing the variance (going closer to the boundary), and also selecting the variable (going down with the arrow):



We are getting a sparser solution than ridge, since the variance is more reduced, allowing more variable selection:



You can do spiker and spiker constraints. Up to the limit of pure variable selection (either take or not take that β).

7-5

Ensemble methods

Given a training set X, with n statistical units with p features, and the target variable $\underline{y}$ that can either $\in R \; (regression)$ or $\in labels \; (classification)$.

The model perform inference as $M_X : R^p \rightarrow \begin{cases} R \; (if \; regression) \\ labels \; (if \; classification \;) \end{cases}$

We must deal with bias variance trade off, always.
We aim to reduce the variance without increasing the bias.

The simplest model to make predictions (take the example of measurement of table with rule) is $y = \mu + \varepsilon$.

If ε is not a systematic error: $E[\varepsilon] = 0$. But there is some variability in the error $Var(\varepsilon) = \sigma^2$.

The simplest model is unbiased. The prediction is random, due to measurement error, that is the variability. Is a characteristic of the "instrument" used for the measurement.

So our aim is to keep the bias we have with the simplest model, but reduce its variance. The straightforward solution is to take several ($say\ B$) independent measurements. In this way we obtain $y_1, y_2, \dots, y_B$ and each $y_i = \mu + \varepsilon_i$.
$E[\varepsilon_i] = 0$ still, and $Var(\varepsilon_i) = \sigma^2$.

Taking the average of measurements $\overline{y} = \frac{1}{B}\sum_{i=1}^{B} y_i$

The mean is still unbiased $E[\overline{y}] = \mu$ and now the **variance is reduced** $Var(\overline{y}) = \frac{\sigma^2}{B}$.
We didn't change the instrument (model), but only the measurement. In this way we reduced variance.

The only prerequisite is to make independent predictions, with independent training sets.

After having built an unbiased model M_X for X.
We must replicate the dataset B times using the same "law" that generated the original dataset X.
$X_1, \dots, X_B$ datasets each of n units.
Fit each dataset with $M_{X_1}, .., M_{X_B}. B\ models.$

Build the ensemble model $M = \frac{1}{B}\sum_{i=1}^{B} M_{X_i}$

If the $X_1, \dots, X_B$ datasets are independent → M is unbiased with reduced variance than initial models.

The main criticality is that we typically don't have that much data to train/test all those models.
Shouldn't be better to use directly all those data for a single model? Yes, but the variance would still be big.

We use the training set for both training the models and generate new data.

The algorithm that implements this idea is called Bootstrap.

Bootstrap

Given a training set X with n units.
We want to generate a new training set $\tilde{X}$, repeat this B times.
$Pseudocode$:
$For\ i = 1, .., n$:
- $random\ sample\ (\boldsymbol{with\ replacement})\ one\ statistical\ unit\ (row)\ of\ out\ X \rightarrow (\underline{x_i}^*, y_i^*)$
- $\tilde{X}.add(\underline{x_i}^*, y_i^*)$

$\tilde{X}$ will have n units, as original dataset.
Note: the same observation can occur more than once in a new training set.
Generating B datasets, and training B models with those generated data to get then the prediction as the average, is called **bagging**.

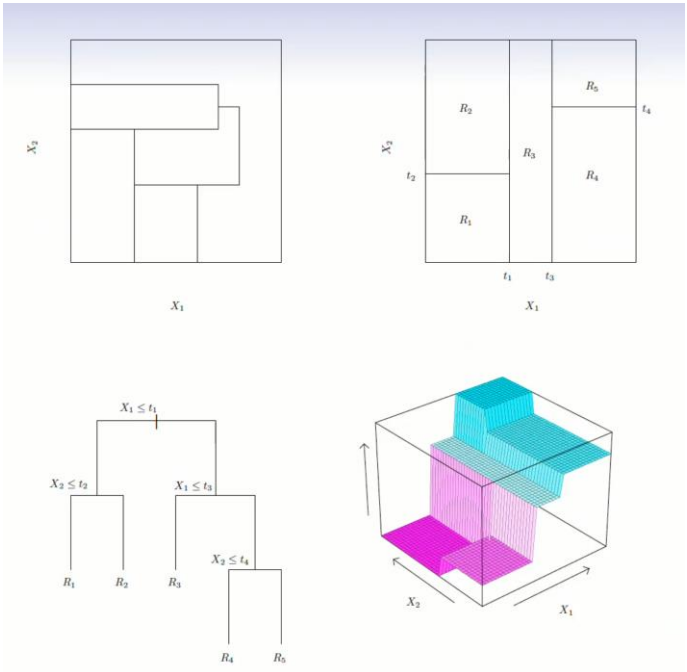
The probability that a unit $u \in X$ that belong to original dataset doesn't belong to $\tilde{X}$ is
$P[u \notin \tilde{X}] = \left(1 - \frac{1}{n}\right)^n$.
For $n \to \infty$, $P[u \notin \tilde{X}] = e^{-1} \cong \frac{1}{3}$

$\sigma^2 > Var(M) > \frac{\sigma^2}{B}$. Hence, we reduce for sure the variability.

Random Forest
We now want the generated dataset to be more independent.

We get again the idea of CARTs to generate datasets, but spicier: at each step, **instead of splitting on all features, randomly select m features to make the split. (if m=p is the same of bagging).**
Remark on the splits: the longer the split of the tree the more the reduction in variability for that feature.

The goal is to find the regions of the split that minimize the variability between the rectangles. The obvious way (overfitting) is to take rectangles of size 1, containing a single point. So we must constraint the amount of rectangles.
With this strategy you can't create the top-left, you can split the feature space only with rectangles.



You can use this to have an idea of which relationship there is between the features: you see if there is some dependency between them and then you include it in the linear model.

You'll most likely reach overfitting with CARTs, so you introduce some penalization for the amount of terminal nodes (leaves). Lots of leaves imply high variance and low bias. The penalization is chosen with cross validation.

To look at the distribution of labels in each rectangle, and maximize one colour, you either use **Gini** index or **Entropy**. They have **small** value **if all everybody has same colour**, **or if there is a uniform distribution of colours** (you don't want this if doing classification).

OOB: Out Of Bag Error Estimation. We expect an overlap of observations in the generated dataset: $\frac{2}{3}$. Indeed $\frac{1}{3}$ of observations of the original will not be part of the bootstrap sample. **Those $1/3$ of observations can be used as test set.** So when you create the bootstrap sample, you keep track of which observations weren't included in the new training set. In this way there is **no need for Cross Validation!**

With random forest we still can interpret the results.

Boosting

You fit a model, you get the residual, then you fit a model on the residual, and so on.
For sure you'll overfit, so you must discount the prediction on the residual, slowing down the learning on the residual.

13-5

Linear Mixed Models

Independency and homoscedasticity is asking too much.
Modelling general dependence and heteroscedasticity among observations concerns the design of the variance covariance matrix.

For repeated measurements we must model dependence between units.
Traditional linear model will be called fixed effect model from now on.
We must partition the residual in another way.

The idea is that **observations within groups are more similar** (maybe also correlated) **than observations between groups**, if we ignore the dependence structure, the parameter estimation is biased.

LMMs decouple the contribution of different kind of dependencies among observations in groups, doing inference taking into account the presence of groups. The **residual will no longer contain the contribution of dependency,** since is modelled now.

LMs:

- analyse data with independent observations and homogenous variance.

We've estimated parameters through OLS so far, however, is not good for more complex linear models (as LMMs), even tough for Fixed effect model were unbiased!
OLS doesn't require the normality assumptions, but only uncorrelated residuals.
If we go through ML estimation for the parameters, the ML parameters are biased. We must go through another way: restricted maximum likelihood. Look for the solution in a restricted subspace, so that the estimator is unbiased and ML.

OLS estimators are equivalent to the REML estimates only in classical LM with independent, homoscedastic errors, NOT for more complex formulations (LMM).
(maximize the likelihood of the residual)

We now relax the homoscedasticity assumption: allow heteroscedastic observations, keeping the assumption that the observations are independent and normally distributed.

Allow heteroscedastic observations

In LM we've used so far $V[y_j] = \sigma^2$, now we'll assume $V[y_j] = \sigma_j^2 = V[\varepsilon_j] = \sigma^2\lambda_j^2$.

$$\varepsilon \sim N(0, \sigma^2 \Lambda\Lambda) \quad where\ \Lambda = diag(\lambda_1, \dots, \lambda_n)$$

So now we have $\boldsymbol{n + p + 1}$ parameters! The model is not identifiable, so we either **constraint the residual variances by assume known variance weights** (as was done with WLS), **or represent variances more parsimoniously as a function of** a small **set of parameters**:

➔ variance function $\lambda(\delta, \mu; v)$ to introduce heterogeneity of variance.

- Assume positive values and continuous and differentiable with respect to δ, that contains a small set of variance parameters, common to every observation.
  v_j is a vector of covariates defining the variance function for observation j.
- $V[y_j] = \sigma^2\lambda(\delta, \mu_j; v_j)$
  It depends on the constant σ^2 but is multiplied by a function that can have any shape, and depends on:
  - $\boldsymbol{\delta}$: contains a small set of variance parameters, **common to all observations**
  - v_j: known covariates defining the **variance function for observation j**
  - μ_j: depends on β_j too.

4 groups of variance functions:

- $\lambda(v)$ known weights
- $< \delta > -- \lambda(\delta, v)$ Variance function depend on δ but not on μ
- $< \delta, \mu > -- \lambda(\delta, \mu; v)$ Variance functions depend on δ and μ
- $< \mu > -- \lambda(\mu; v)$ Variance functions depend on μ but not on δ

In $< \mu > and < \delta, \mu >$ the β are shared by the mean and variance structures, as opposed to $< \delta >$ group, where we know the weights or variance function.

REML: maximize the likelihood with respect to the β, then we retrieve the maximum likelihood with respect to the σ^2. The advantage of the function is that depends on fewer parameters, optimizing in lower dimension space. Is an approximation.

Allow correlation

Introduce the correlation structure in the variance function.
For instance, studies that collect measurements over time, or per individual, or hierarchical so to lead to correlated data. We don't have independence between units anymore.

Each group will now have a different design matrix Z_i, so $E[y_{ij}] = z_{ij}^T\beta$ and $V[y_i] = V[\varepsilon_i] = \sigma^2 R_i$.
Each group with n_i observations.

We must introduce constraint, since we have more parameters that observations.

- ➔ $Variance\ covariance\ matrix\ for\ group\ i$: $\boldsymbol{R_i} = \sigma^2 R_i = \boldsymbol{\sigma^2 \Lambda_i C_i \Lambda_i}$) where $\Lambda_i = (\lambda_{i1}, .., \lambda_{in_i})$ is allowing for **heteroschedasticity** of observations **within group $\boldsymbol{i}$** (since are the variance function introduced before) and also **allowing correlation of observations within group $\boldsymbol{i}$ with $\boldsymbol{C_i = corr\ matrix}$**

Correlation function is a function of a distance and covariance parameter.
Correlation structure can be classified into two groups:

- Serial structure: time series like, autoregressive
- Spatial structure

Estimates with WLS or likelihood-based methods mentioned before.

Linear Mixed Models

Up to now we modelled dependence among observations acting on the var-cov matrix of the errors.
The issue with LM3.0 is that we must assume the exact kind of dependence.
Estimate (or model) separately both the **group level effect and dependence between observations** not derived from the var-cov matrix of the errors but **driven by assumptions on between groups variability**.
Partition the global variance, and inference on the population of the groups.

AIC definition

$AIC = 2\mathrm{k} - 2\ln(\mathrm{L})$ where k is the number of parameters and L is the maximized estimated likelihood.

BIC definition

$AIC = \mathrm{k}\ln(\mathrm{n}) - 2\ln(\mathrm{L})$ where k is the number of parameters L is the maximized estimated likelihood and n is the number of data points.