Clustering of neutral (γ, π^0 and ν initiated shower) particles

SHILPI JAIN(TIFR)

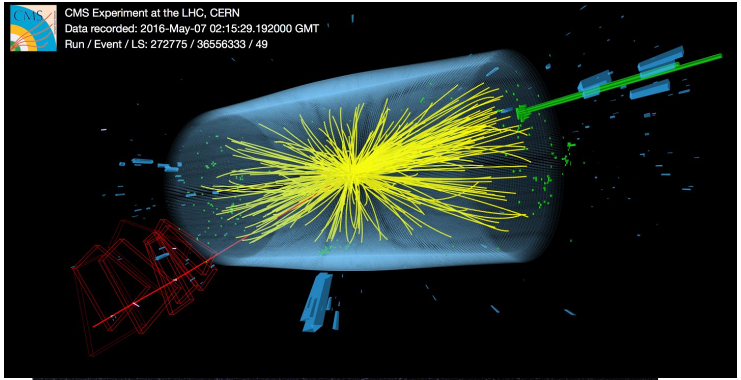
EHEP2026 School, TIFR Mumbai

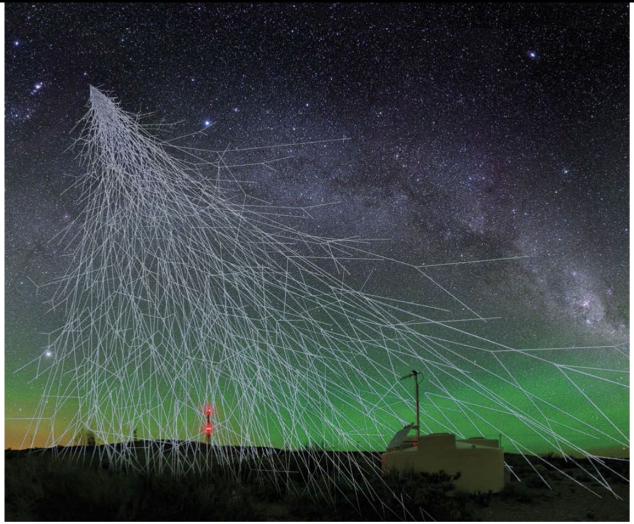
16th - 23rd Feb, 2026

An important step in understanding fundamental physics

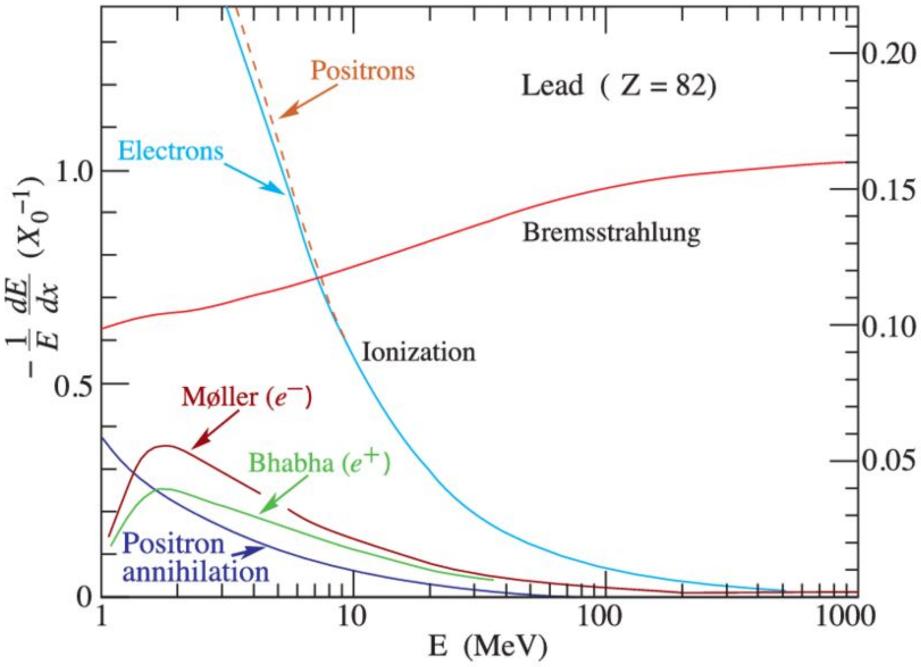
- As high energy physicist, we are always looking for new physics/new sources or testing fundamental physics
- To make any such measurement, an intermediary step is the reconstruction of particles that have left marks in the detector
- By reconstruction of particles, we mean, information about
  - Energy, momentum
  - Position
  - Direction
  - Particle type etc

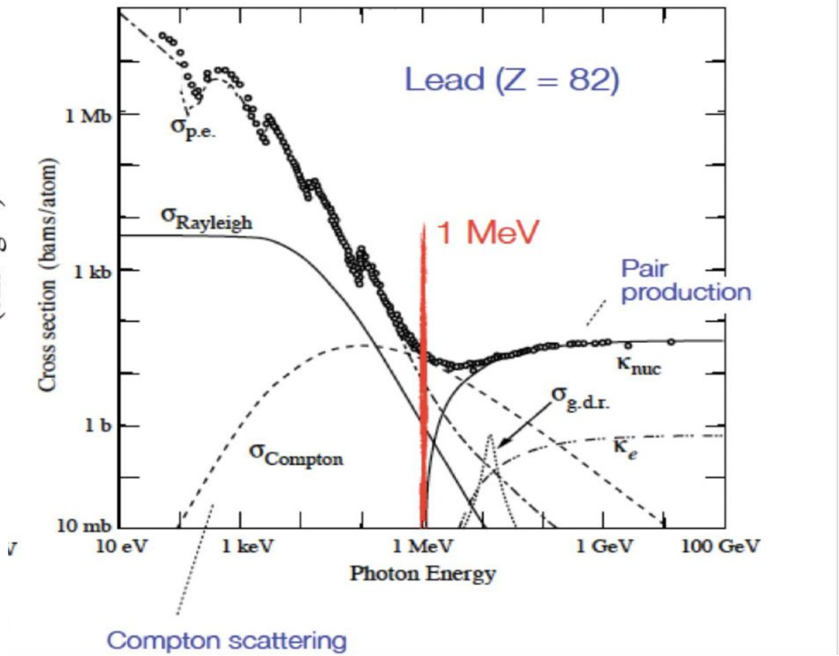
Important to understand how different particles interact in the detector





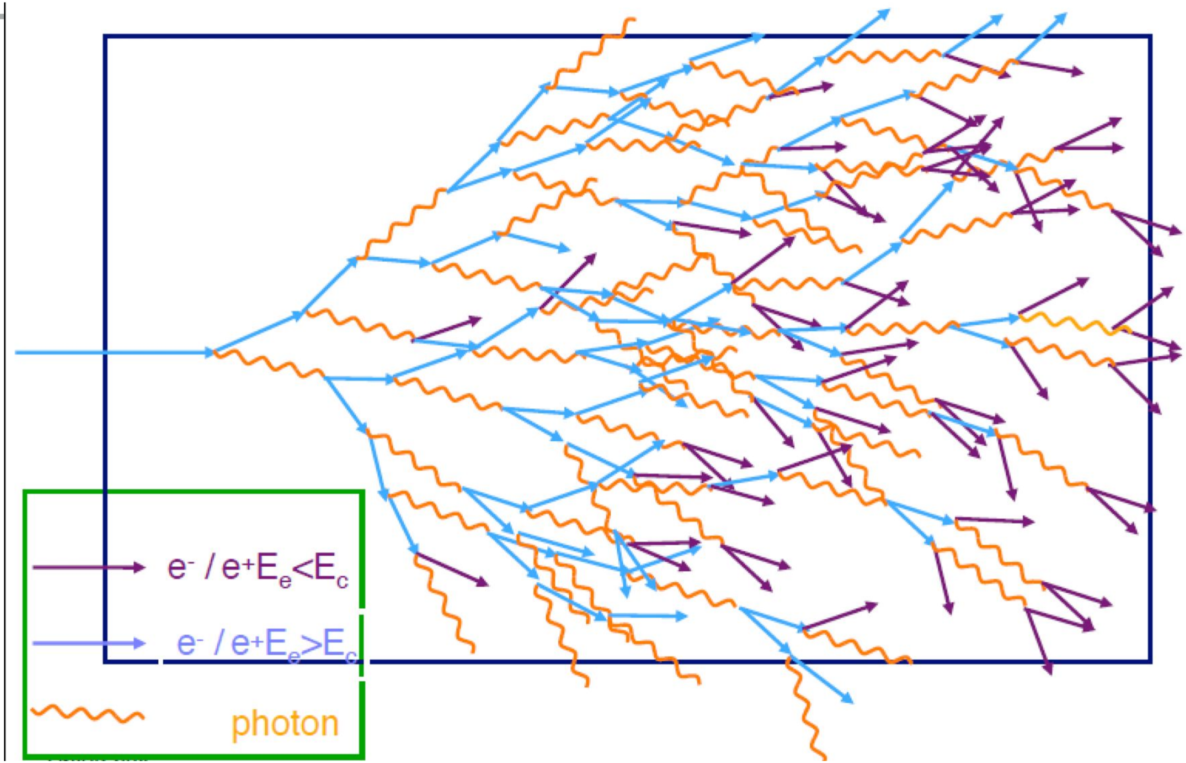
Ways particles interact with matter





- Which process takes place depends on the cross-section
- We will remain in the high energy regime, i.e., electrons and photons energy in GeV

In GeV regime, how do particles deposit energy

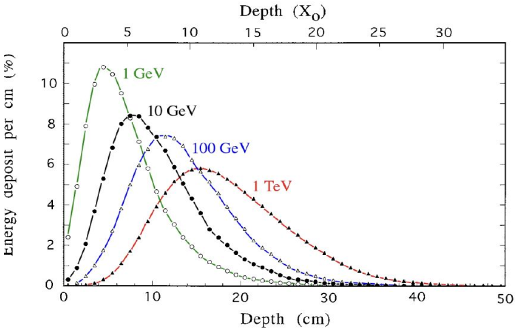


Credits: Unknown

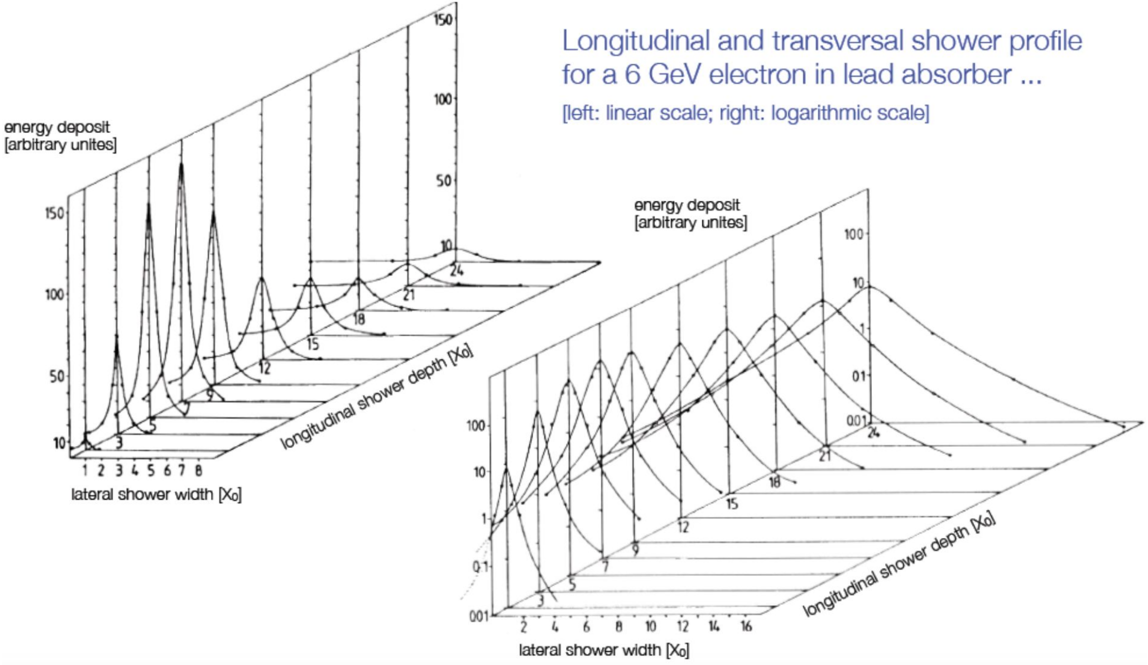
- Which process takes place depends on the cross-section
- We will remain in the high energy regime, i.e., electrons and photons energy in GeV

Longitudinal and transverse shower profiles

$$t_{max} = 1.45 \ln(E_0/E_c)$$



Electron shower in a block of copper

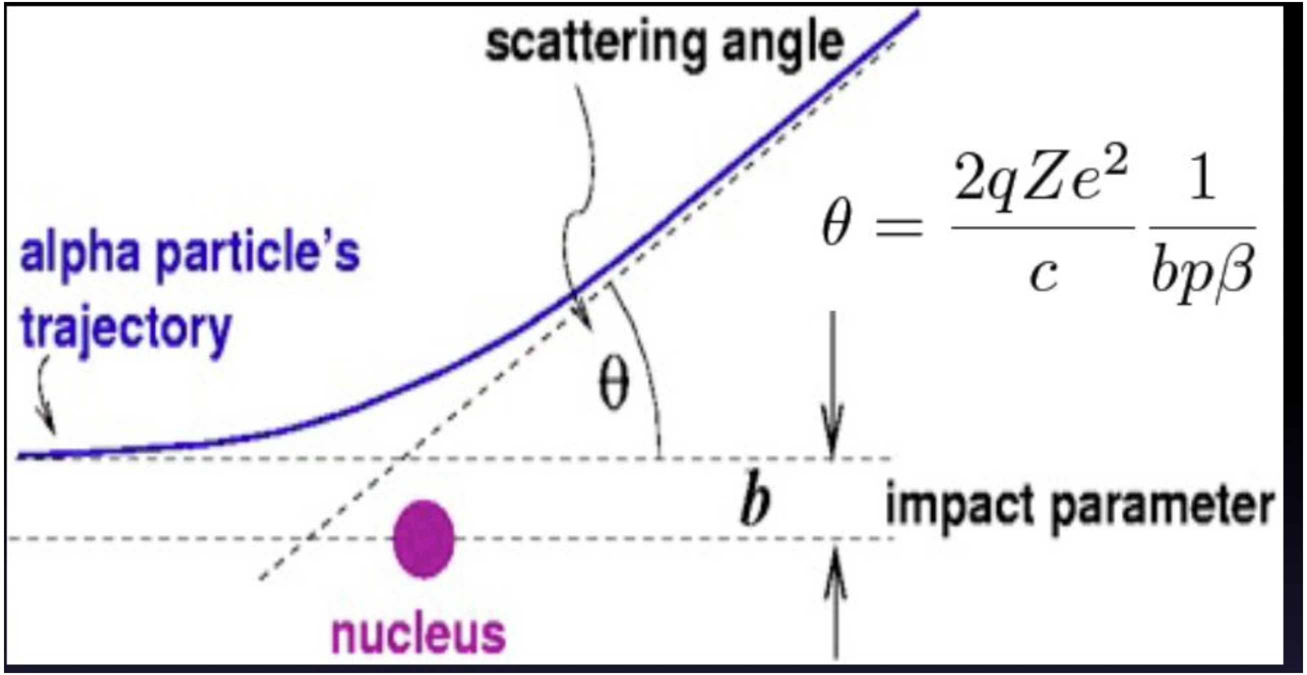


Longitudinal and transversal shower profile for a 6 GeV electron in lead absorber ...

[left: linear scale; right: logarithmic scale]

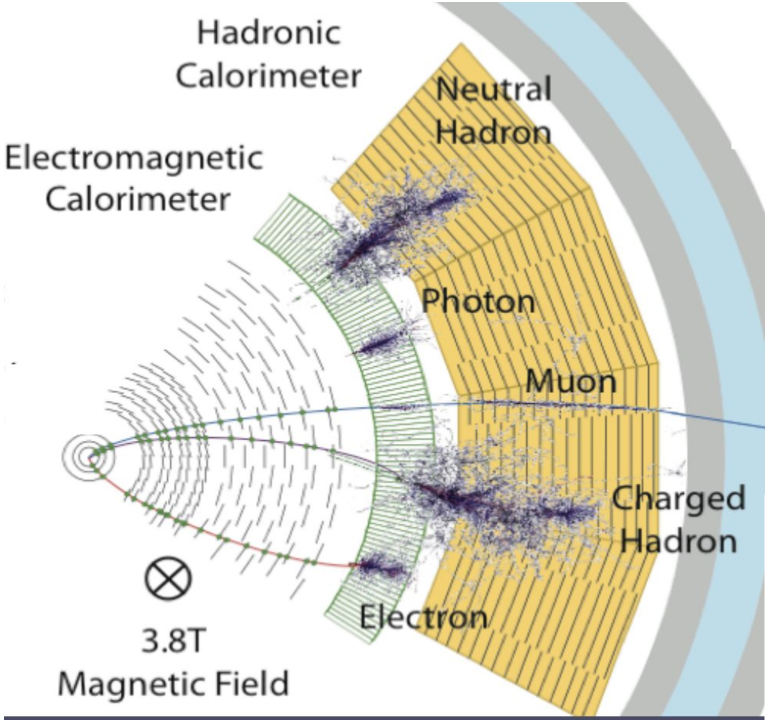
- Transverse shower spread is entirely due to multiple scattering

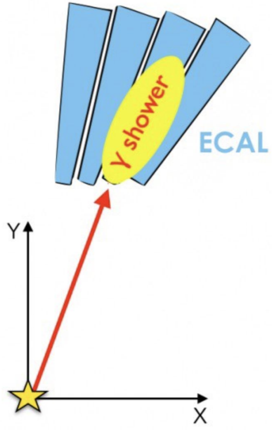
Reminder about multiple scattering

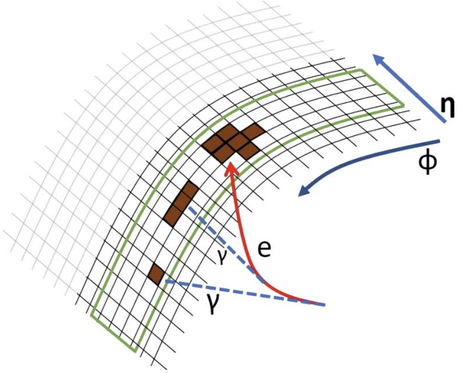


- A particle passing through material undergoes multiple deflections due to Coulomb scattering with the nuclei

Typical signature of high energy electron and photon in a detector



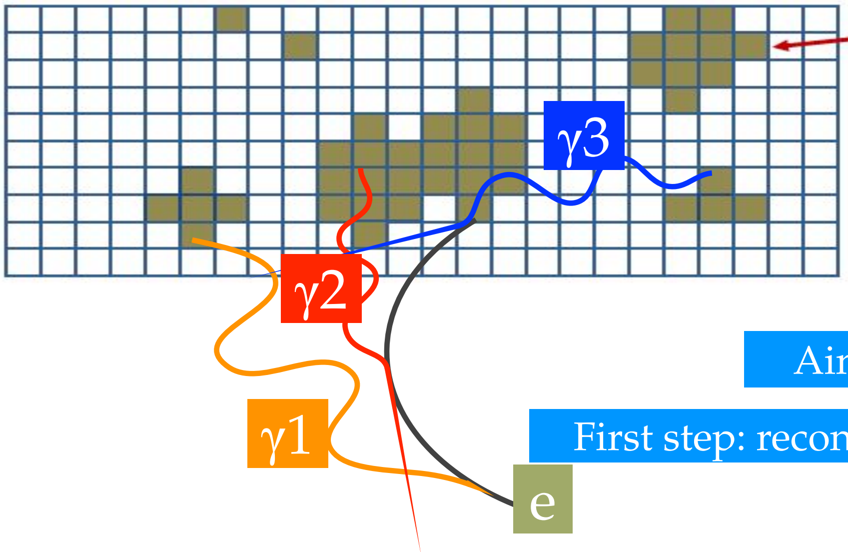


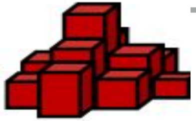


- Presence of magnetic field and material before the calorimeter can further cause complicated scenarios due to bremsstrahlung and pair production

ELECTRON AND PHOTON SHOWERS

Sam Harper





the rec-hits are not of uniform energy
have a peaked profile

Aim is to reconstruct this full energy

First step: reconstruct individual energy deposition

- An electron or a photon deposits energy in several crystals
- Aim: reconstruct the full energy of electron or a photon
- Once we have the crystal energies reconstructed, we need to reconstruct individual particles from deposit of a single photon/electron
  - i.e. in this picture reconstructing individual energy deposits from γ1, γ2, γ3 and the final electron
  - this is the PF (particle flow) clustering step - i.e. making small clusters
  - These small clusters are then combined to form super-clusters

Clustering

- Clustering is needed to collect all the imprints of a particle in that sub-detector to reconstruct the particle's energy, momentum, direction etc

Clustering algorithms in collider experiments

ISLAND CLUSTERING

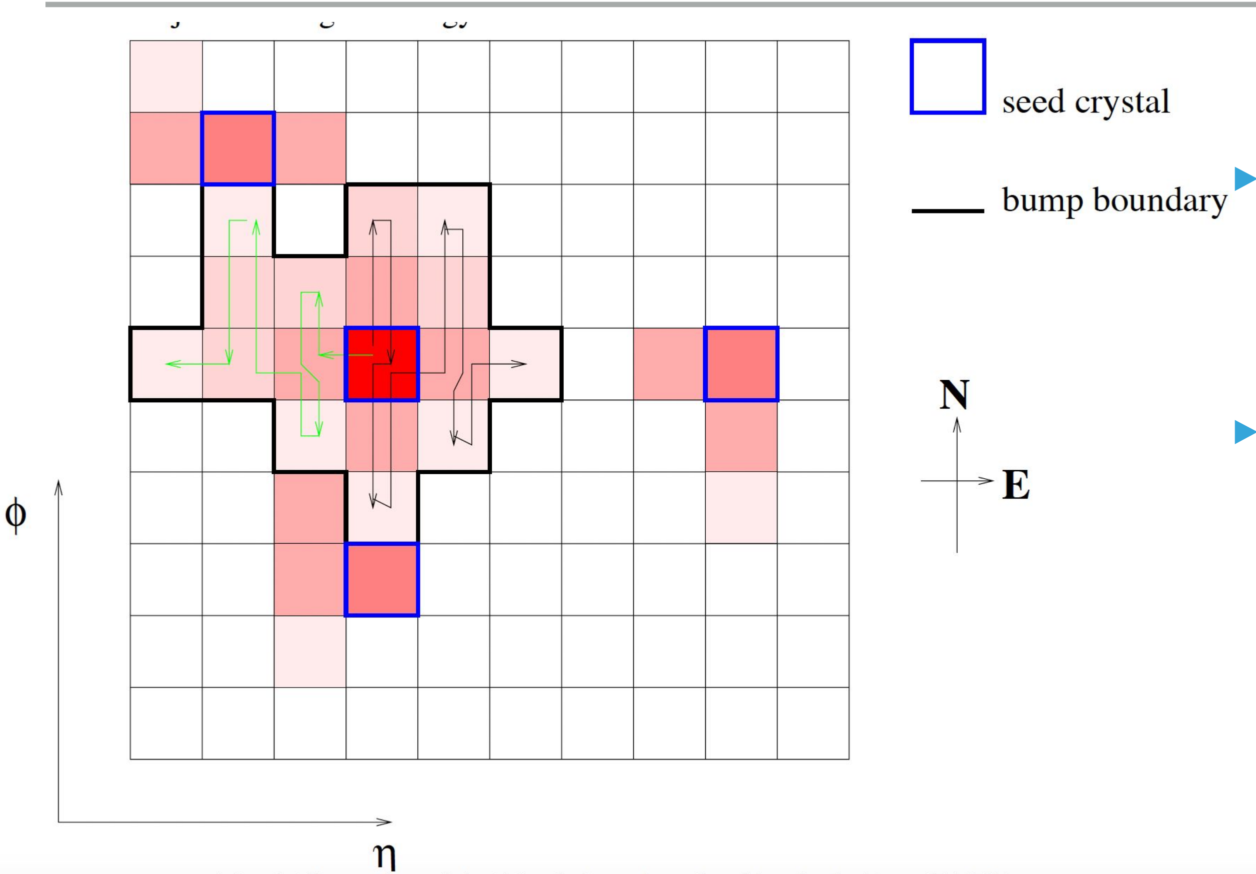
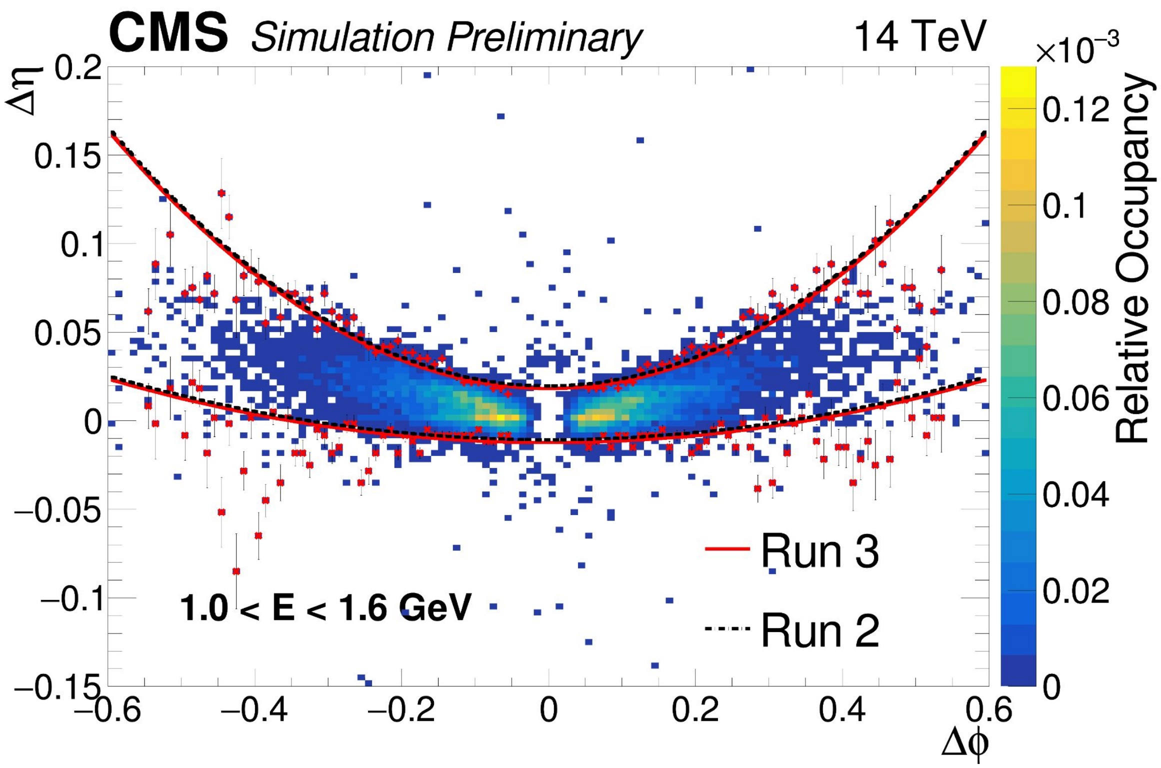


Fig. 1: Illustration of the Island clustering algorithm in the Barrel ECAL

- Start with maximum energy crystal —> seed
- Go in all the directions and collect hits as long as the hit energy > noise threshold and there is no bump in energy

Collecting the clusters

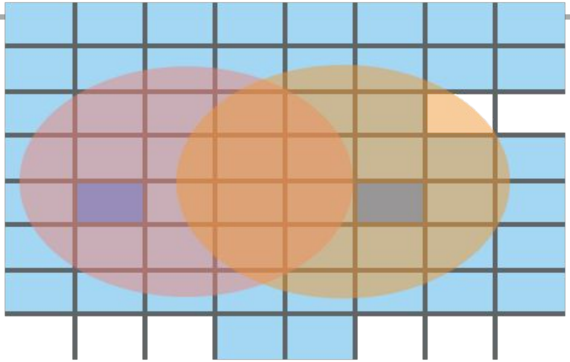


Clustering of clusters depends on how the clusters are placed on the detector surface

Collect within the parabolas

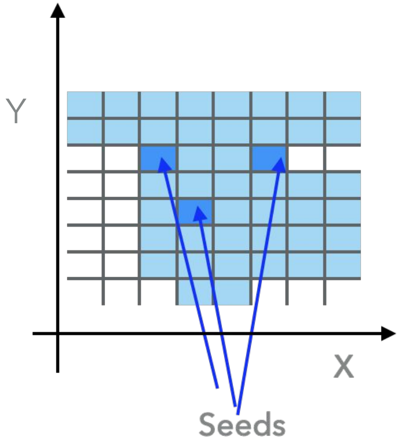
What if the showers from two particles are merged? This algorithm does not work (as we will see in the exercises also)

PF CLUSTERING I.E. RECHITS TO SMALL CLUSTERS



Aim: Reconstruct clusters of energy deposit from the individual particles

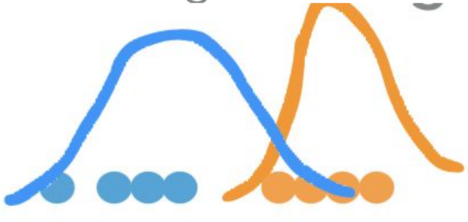
- A cluster can have energy contribution from N photons or an electron and a photon etc.
- It is important to separate the individual contributions.
- **Aim is to cluster the hits coming from the same particle and also determine the fraction of energy associated to a particle if a hit is shared**
- If there are M crystals in a cluster with N seeds, then these N seeds are coming N particles. The energy spread in other xtals from a particle is modeled using Gaussian, so N Gaussians
  - where seeds are crystals with local energy maxima above a threshold and are identified w.r.t the neighboring cells (8)
- For this purpose, an algorithm named **expectation maximization (EM) based on gaussian mixtures** is used



PROBLEM AT HAND

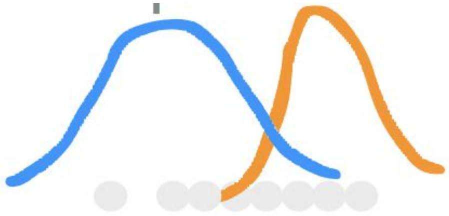
- Belongs to the class of clustering methods for ‘soft clusters’ in ML language (i.e. assigns probability)

- Problem 1: **Given a set of points from two gaussian**, we can evaluate the mean and sigma of the underlying Gaussians. $\mu = (x1 + ... + xn)/n$



Straightforward

- Problem 2: Given a set of points, and known parameters of two gaussian, we can tell from which Gaussian those point most likely belong using Bayes’ theorem. Example if there are two gaussians ‘a’ and ‘b’ and point xi
  - $P(a|xi) = P(xi|a) * P(a)/(P(xi|a) * P(a) + P(xi|b) * P(b))$



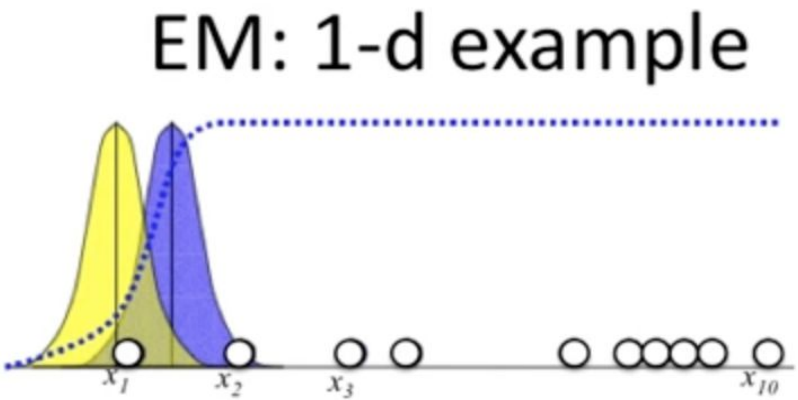
PROBLEM 3

- Problem 3: Given a set of points, and no other information, we need to construct the gaussian distribution - relevant to our case
  - In our case, we do not know the mean and sigma of the gaussian
  - Use expectation-maximization algorithm
    - In the first step, calculate posterior probability (P(a|xi)) of each point from each gaussian starting with some initial values of mean and sigma. In the next step, update the gaussian parameters using the knowledge from the first step

Set of points/xtals

Determine:
What xtals belong together
and what the parameters
of the gaussians?

STEP I: EXPECTATION STEP



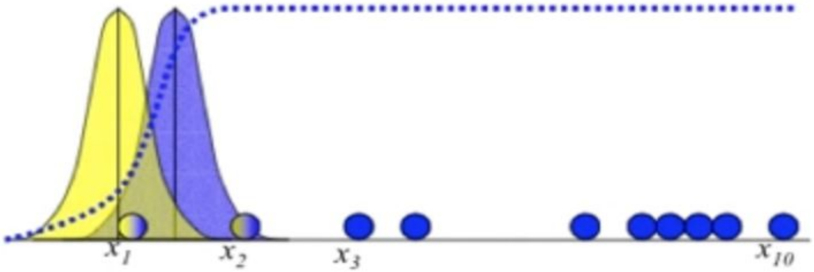
$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

- Start with some initial values of means (μ) and sigma (σ) of two gaussians ‘a’ and ‘b’
  - Calculate the probability that given Gaussian ‘a’ with mean μ_a and σ_a, what is the probability of getting point x_i for Gaussian ‘a’
  - Same for Gaussian ‘b’

STEP 2: MAXIMIZATION STEP

EM: 1-d example



$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

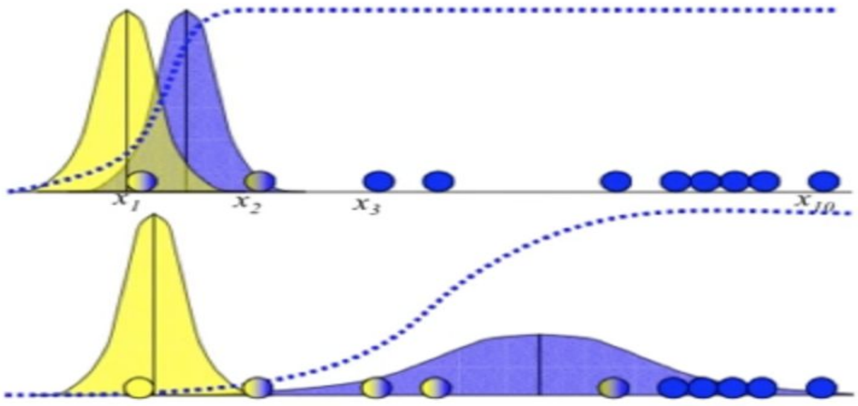
$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \ldots + b_n x_n}{b_1 + b_2 + \ldots + b_n}$$

$$\sigma_b^2 = \frac{b_1(x_1 - \mu_b)^2 + \ldots + b_n(x_n - \mu_b)^2}{b_1 + b_2 + \ldots + b_n}$$

- For Gaussian ‘a’, calculate μ_a and σ_a again using these estimated probabilities by summing over all the hits
- Same for Gaussian ‘b’
- Now you have updated μ_a and σ_a for Gaussian ‘a’ and μ_b and σ_b for Gaussian ‘b’

STEP 3

EM: 1-d example



$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left\{-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right\}$$

$$b_i = P(b \mid x_i) = \frac{P(x_i \mid b)P(b)}{P(x_i \mid b)P(b) + P(x_i \mid a)P(a)}$$

$$a_i = P(a \mid x_i) = 1 - b_i$$

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \ldots + b_n x_n}{b_1 + b_2 + \ldots + b_n}$$

$$\sigma_b^2 = \frac{b_1(x_1 - \mu_b)^2 + \ldots + b_n(x_n - \mu_b)^2}{b_1 + b_2 + \ldots + b_n}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \ldots + a_n x_n}{a_1 + a_2 + \ldots + a_n}$$

$$\sigma_a^2 = \frac{a_1(x_1 - \mu_a)^2 + \ldots + a_n(x_n - \mu_a)^2}{a_1 + a_2 + \ldots + a_n}$$

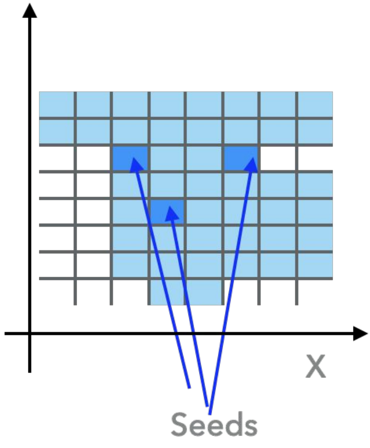
- With these updated μ_a and σ_a , go to step 1
- Repeat both the steps for let's say 100 iterations

Illustration plotted using an adapted code from Satyaki Bhattacharya : here

EXPECTATION-MAXIMIZATION ALGORITHM

- Consider a total of M cells with N seeds
  - N particles —> N gaussians
- Aim essentially reduces to determining the parameters of the model are μ (space coordinate of the gaussian), and A (amplitude of that gaussian)
- **Expectation step**: Fraction of energy in a cell due to a gaussian is given by
  - where j is the j^{th} cell and i is the i^{th} gaussian. μ is the space coordinate of the gaussian and c is the coordinate of the cell in question
-

$$f_{ji} = \frac{A_i e^{-(\vec{c}_j - \vec{\mu}_i)^2/(2\sigma^2)}}{\sum_{k=1}^{N} A_k e^{-(\vec{c}_j - \vec{\mu}_k)^2/(2\sigma^2)}}.$$



EXPECTATION-MAXIMIZATION ALGORITHM

- **Maximization step**: the parameters of the model (which are A and μ) are updated in the next step using the information from the first step

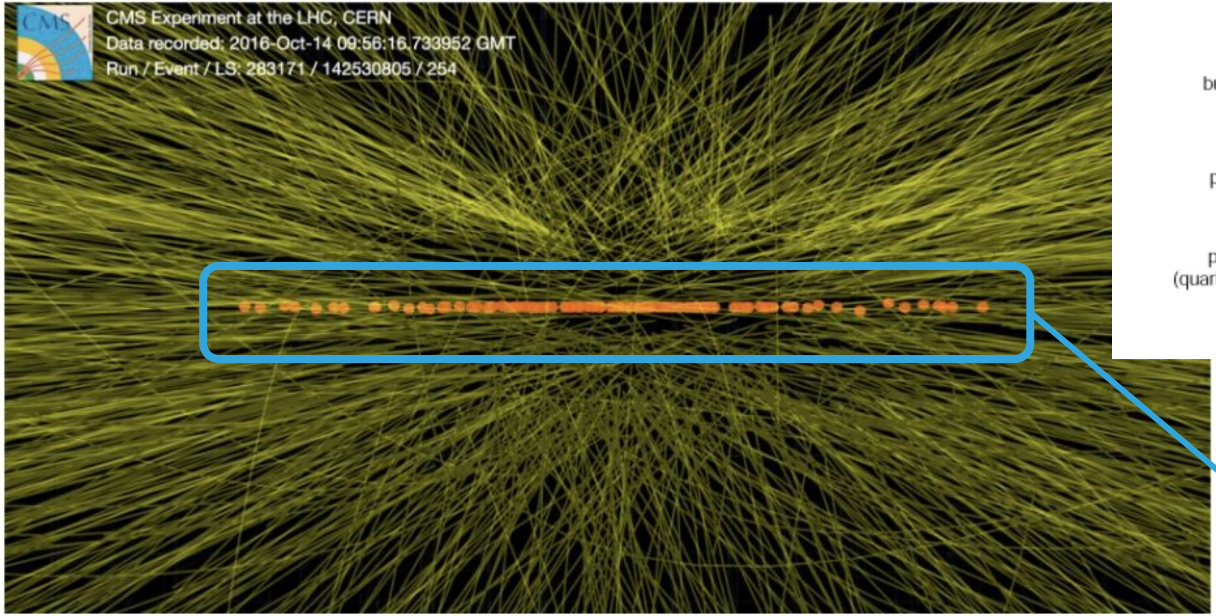
$$A_i = \sum_{j=1}^{M} f_{ji} E_j,$$

$$\vec{\mu}_i = \sum_{j=1}^{M} f_{ji} E_j \vec{c}_j.$$

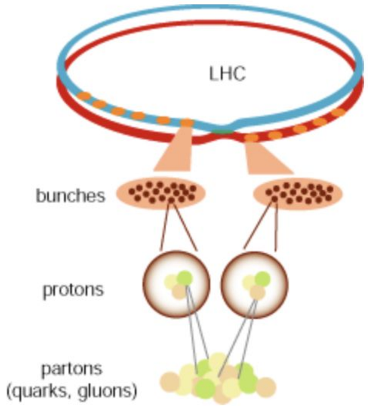
Illustration plotted using an adapted code from Satyaki Bhattacharya : here

Clustering in even messier environments: HL-LHC

Environment in HL-LHC



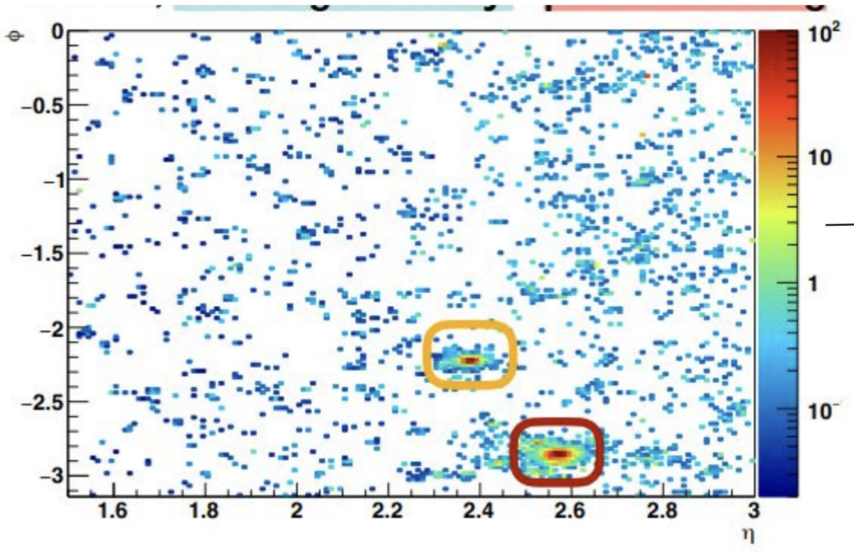
A typical event from a 2016 High PU (<μ> = 100) run

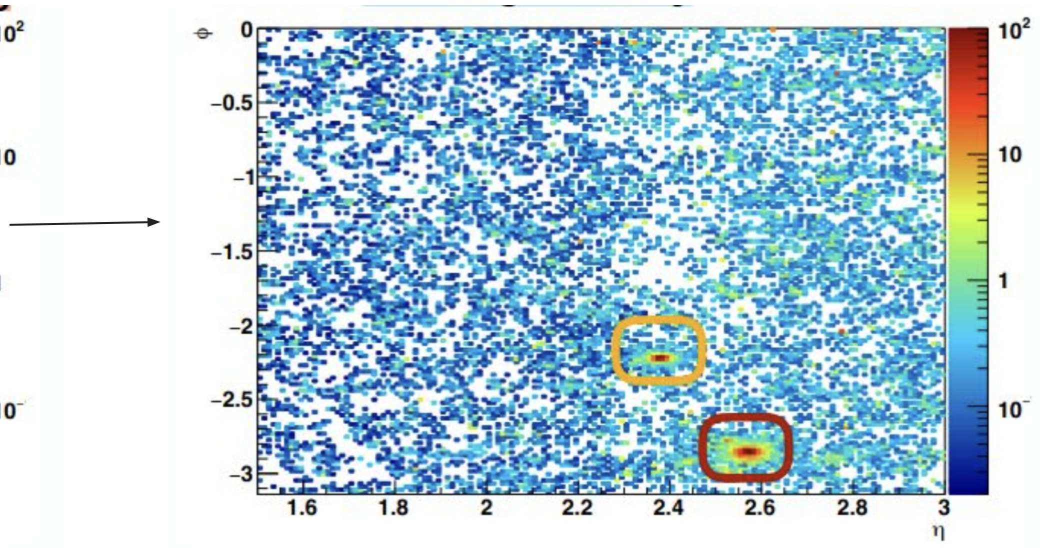


Different proton-proton collisions in 1 bunch crossing

- You can imagine all the particles from all the

Additional particles VS no additional particles



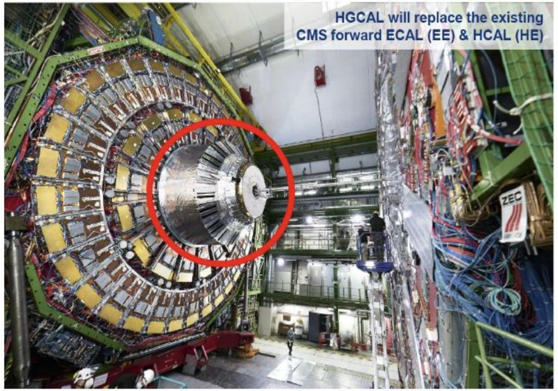


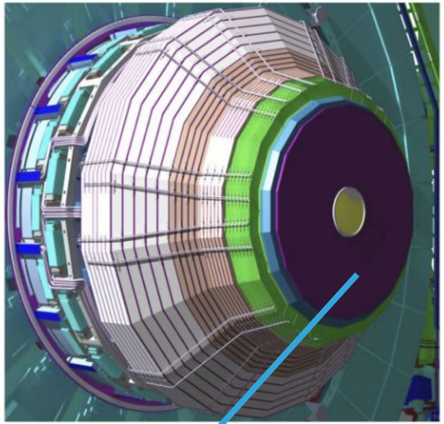
Two photon clusters when the pile up is small

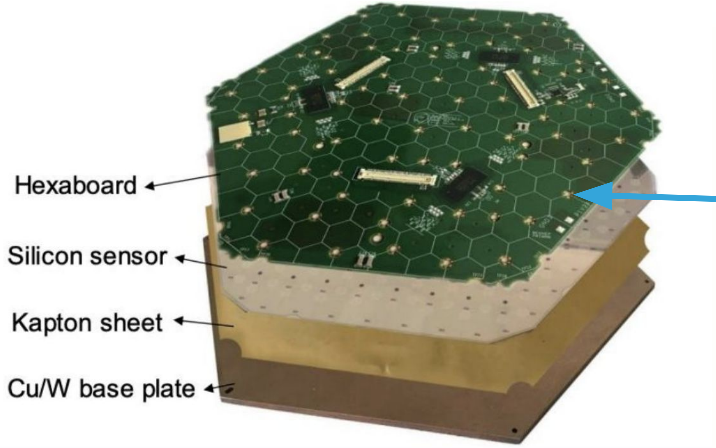
Two photon clusters when number of pp collisions = 200

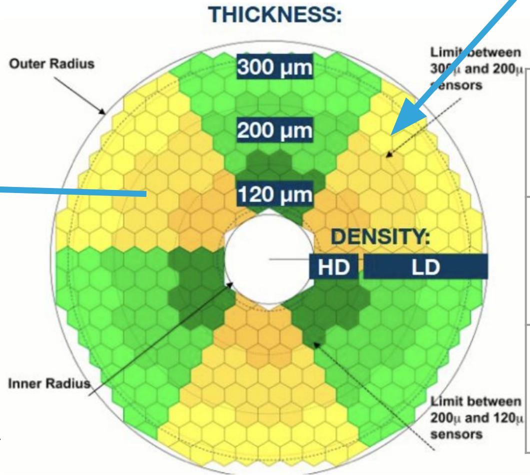
- In the presence of huge pile-up we may end up clustering even those hits which do not belong to the photon → worse energy resolution

High Granularity Calorimeter

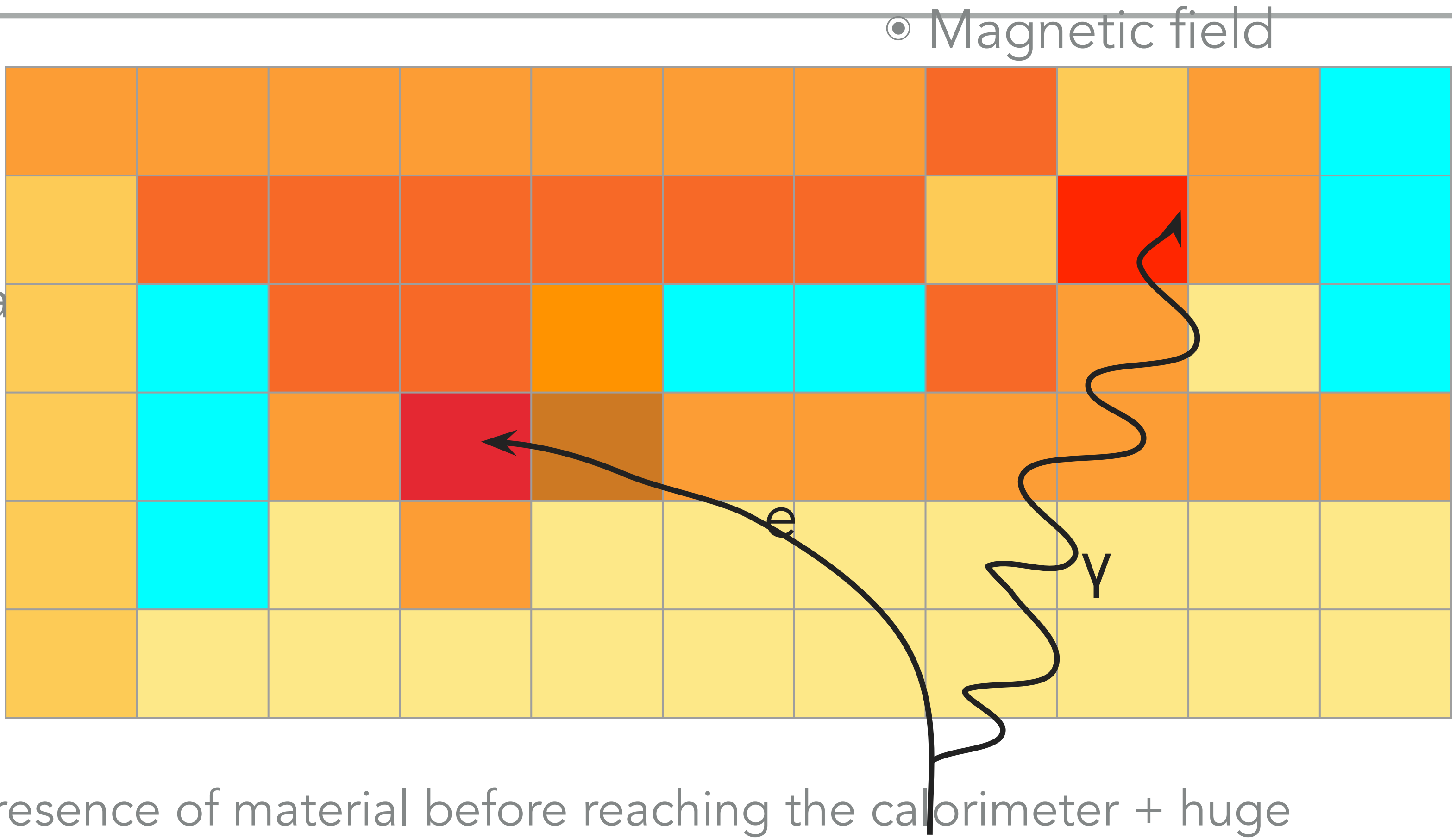








Parameter	Current ECAL	Foreseen HGCAL
Cell size	2.2 x 2.2 cm^2	0.5 - 1.1 cm^2
Number of channels	~72k	6.2 M



Clustering becomes a problem!

In the presence of material before reaching the calorimeter + huge pile-up

Photon should be tangent to the electron line → could not do better

Challenges in Clustering

- Pileup particles should not be clustered
- Computational challenge
  - Large data: Number of channels is ~6.2 M
  - Time constraint: Everything has to be done within ms level
- Clustering algorithm should be highly efficient with minimal computational complexity.
- Preferable to have a fully parallelizable clustering algorithm that can run on for e.g. GPUs, achieving a higher event throughput and a better energy efficiency.
- Keeping all this in mind, a clustering algorithm that uses the idea of density-based clustering was developed

Clustering by fast search and find of density peaks
Alex Rodriguez and Alessandro Laio
Science **344**, 1492 (2014);
DOI: 10.1126/science.1242072

Developers of this algorithm

MACHINE LEARNING

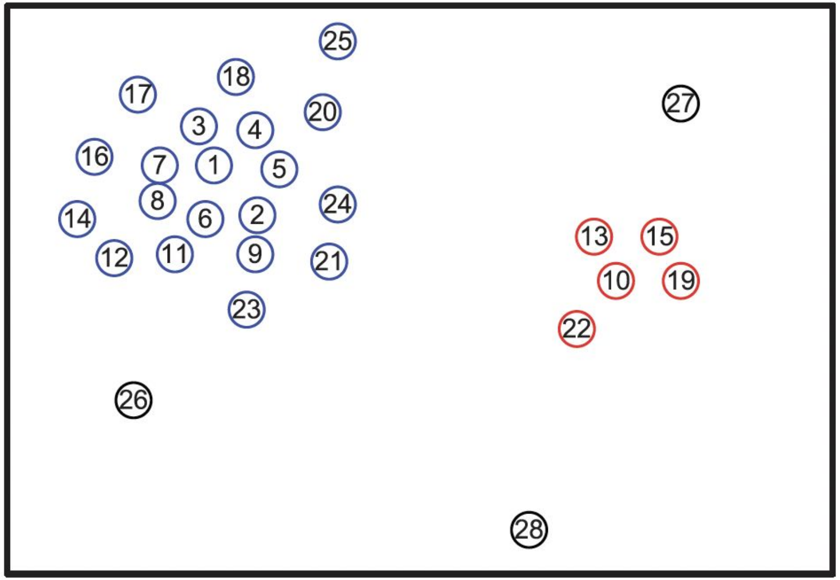
Clustering by fast search and find of density peaks

Alex Rodriguez and Alessandro Laio

Cluster analysis is aimed at classifying elements into categories on the basis of their similarity. Its applications range from astronomy to bioinformatics, bibliometrics, and pattern recognition. We propose an approach based on the idea that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities. This idea forms the basis of a clustering procedure in which the number of clusters arises intuitively, outliers are automatically spotted and excluded from the analysis, and clusters are recognized regardless of their shape and of the dimensionality of the space in which they are embedded. We demonstrate the power of the algorithm on several test cases.

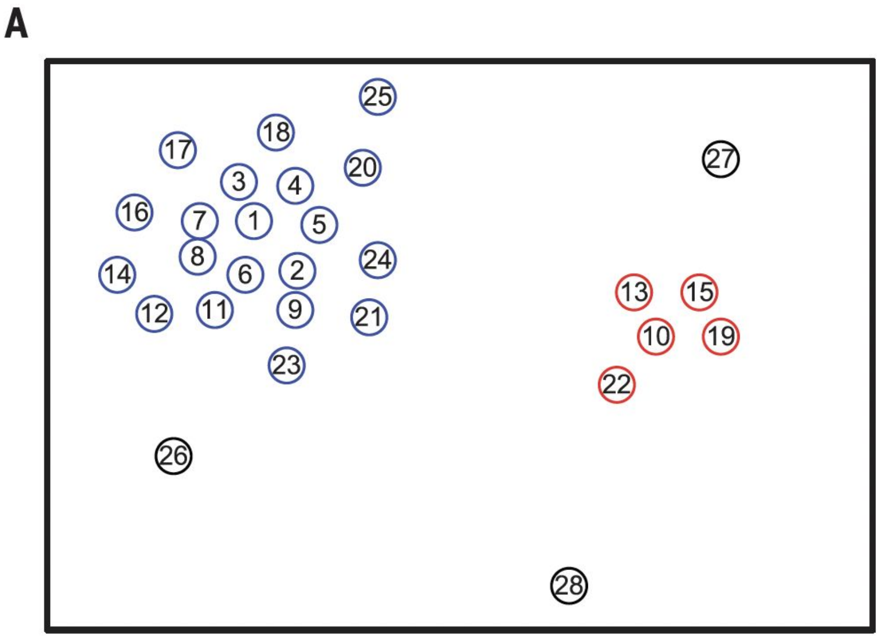
Density based Clustering

A



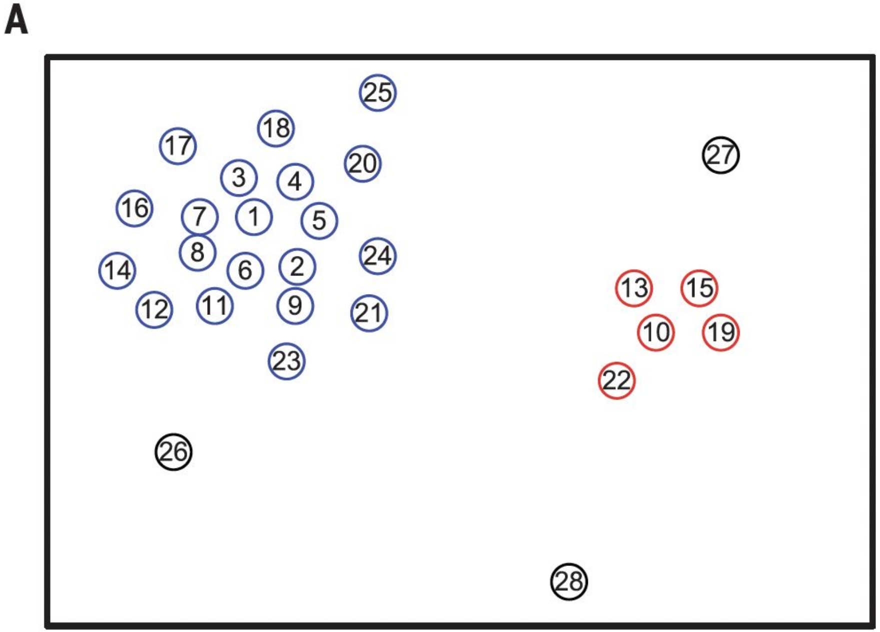
- Intuition behind density based clustering
  - Clusters are surrounded by neighbours with local density
  - They are at a relatively larger distance from any points with a higher local density

Density based Clustering



- E.g. we can see two clusters
- Points (1), (6), (8) and similar are surrounded by low dense points e.g. (25), (20)
- Also, points (1), (6), (8) and similar are far away from points (10), (19) and similar which are another set of high density points
- Using this idea, an algorithm is set in place

Density based Clustering: Algorithm



Core Idea

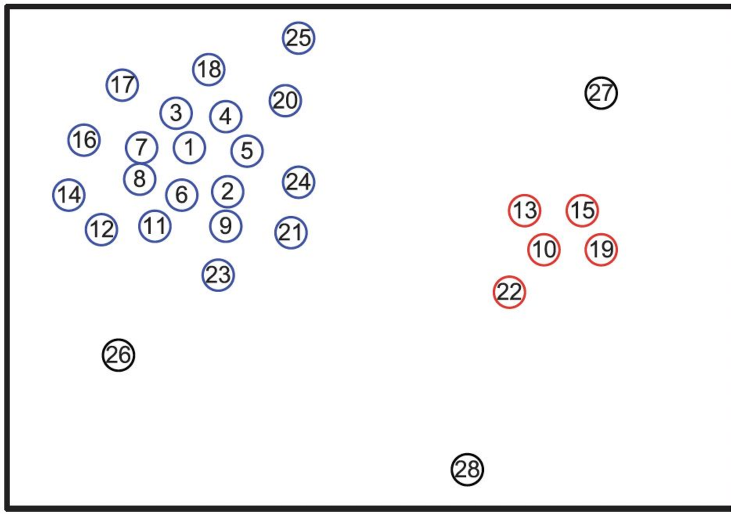
Cluster centers are points that:

- Have high local density ρi
- Are far from any point with higher density δi

Only pairwise distances dij are required.

Density based Clustering: Algorithm

A



Essentially counts the number of points within d_c distance

- Choice of d_c : based on shower lateral size

Step 1 — Compute Local Density ρ_i

For each data point i:

$$\rho_i = \sum_j \chi(d_{ij} - d_c)$$

where

$$\chi(x) = \begin{cases} 1 & x < 0 \\ 0 & \text{otherwise} \end{cases}$$

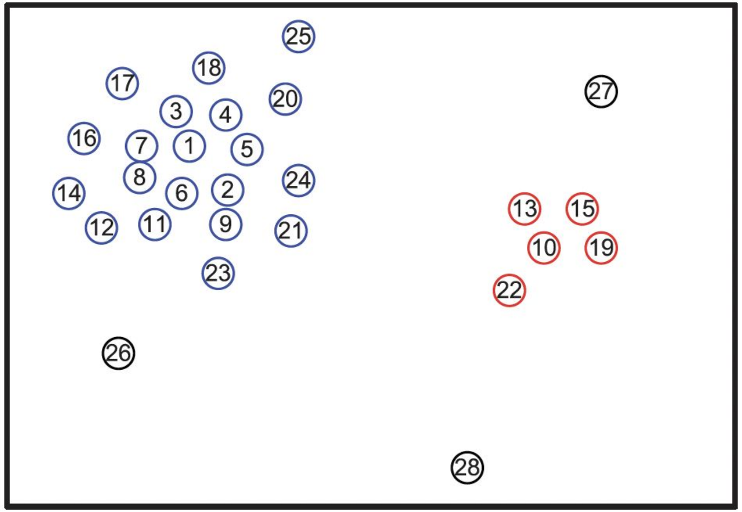
- d_c = cutoff distance
- ρ_i = number of points within distance d_c of point i

Interpretation:

- Counts neighbors inside a sphere of radius d_c
- Only relative magnitude of ρ_i matters

Density based Clustering: Algorithm

A



Step 2 — Compute Distance to Higher Density δ_i

For each point i:

$$\delta_i = \min_{j:\rho_j > \rho_i} d_{ij}$$

Interpretation:

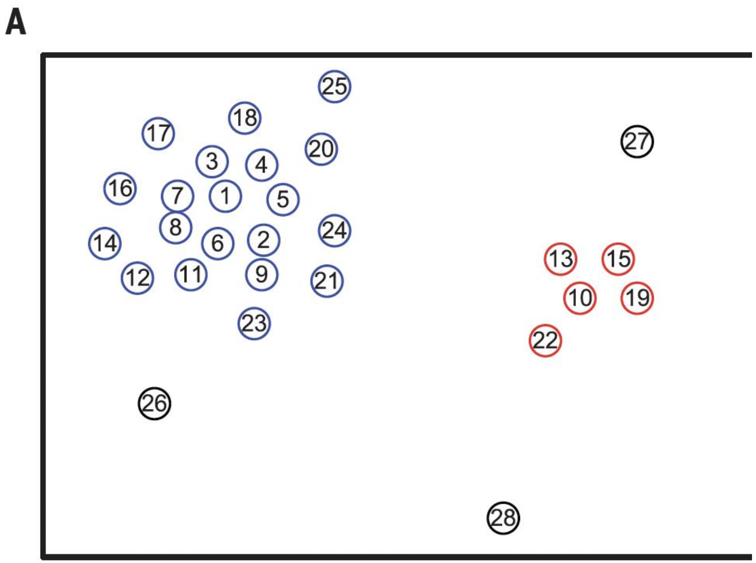
- Distance to the nearest point that has higher density
- Measures how isolated a density peak is

Special case:

For the point with **maximum density**:

$$\delta_i = \max_j d_{ij}$$

Density based Clustering: Algorithm



Step 3 — Identify Cluster Centers

Plot the **decision graph**:

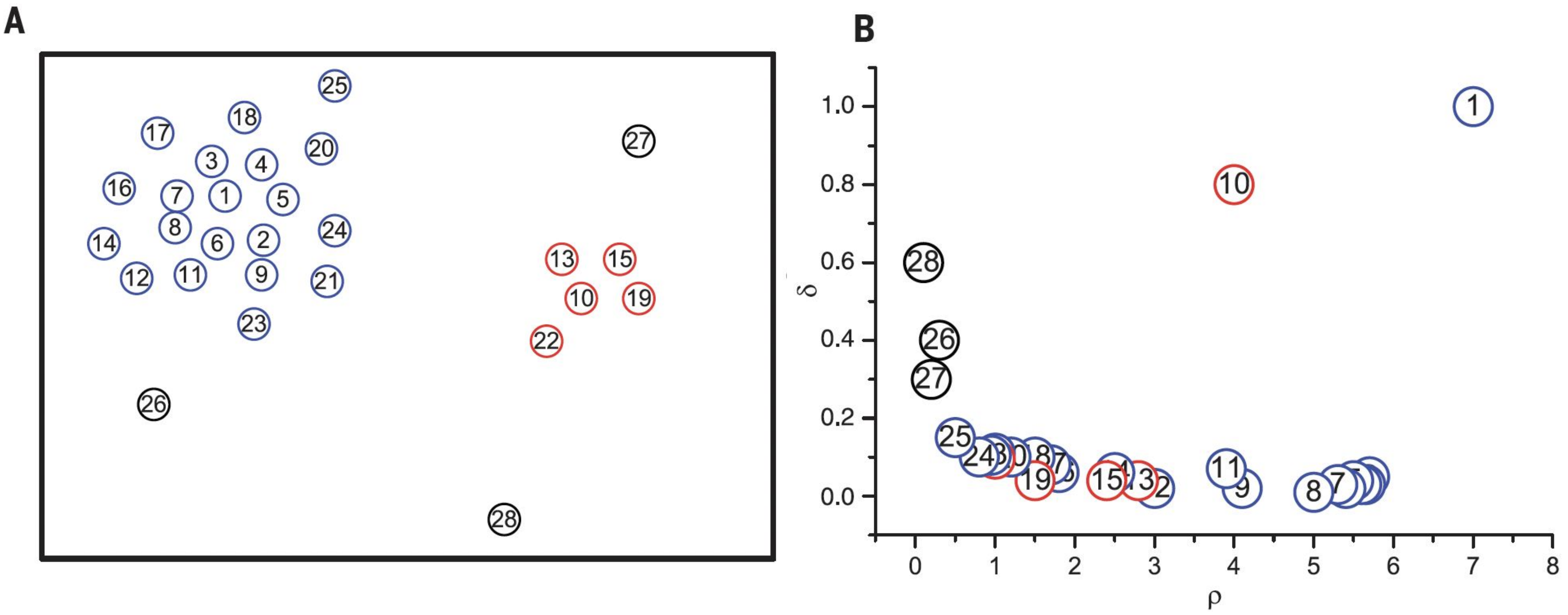
$$(\rho_i, \delta_i)$$

Cluster centers:

- Large ρ
- Large δ

These appear as outliers in the upper-right region.

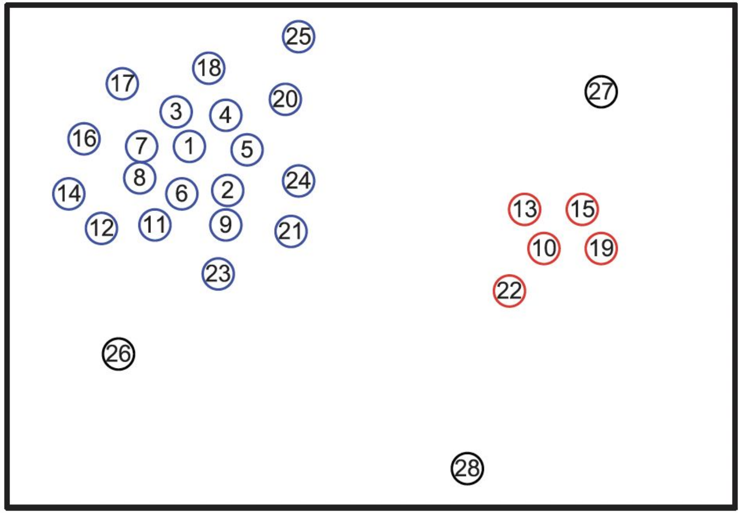
Density based Clustering



From this decision graph, how to identify the clusters?

Density based Clustering: Algorithm

A



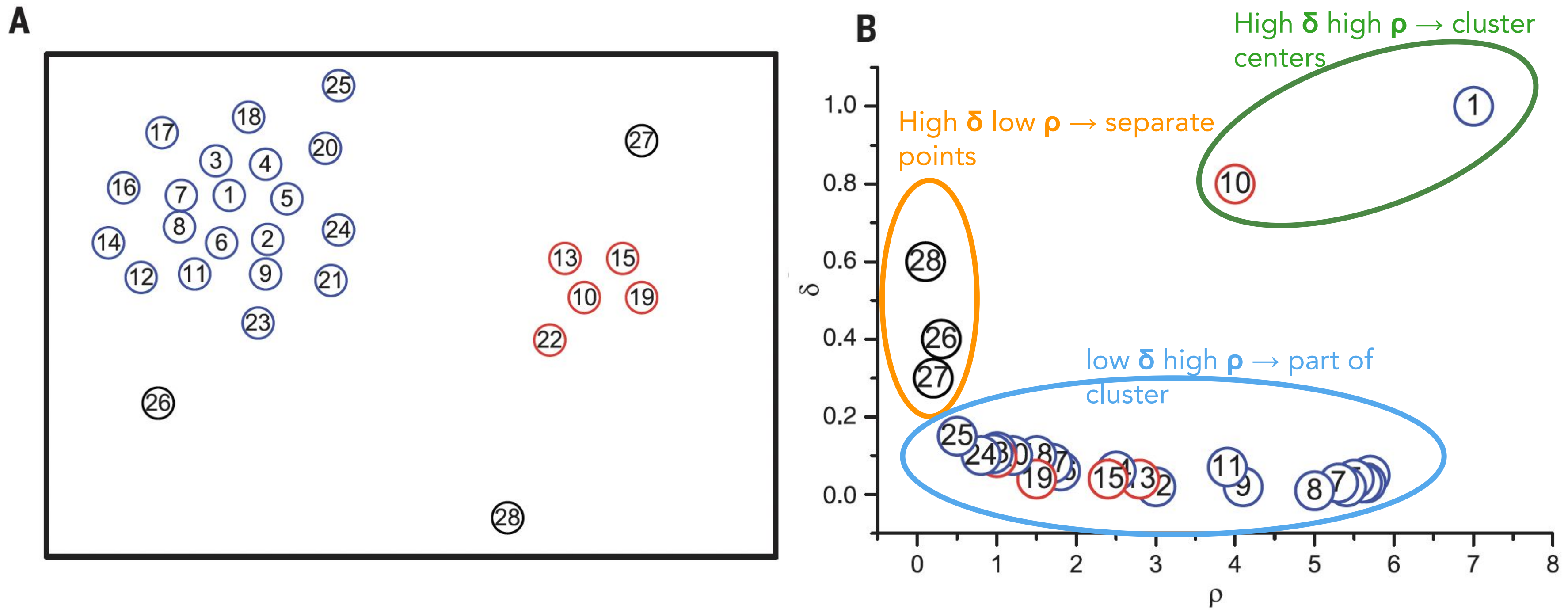
Step 4 — Assign Remaining Points

For each non-center point:

- Assign it to the **same cluster as its nearest neighbor of higher density**

This creates a natural gradient flow toward density peaks.

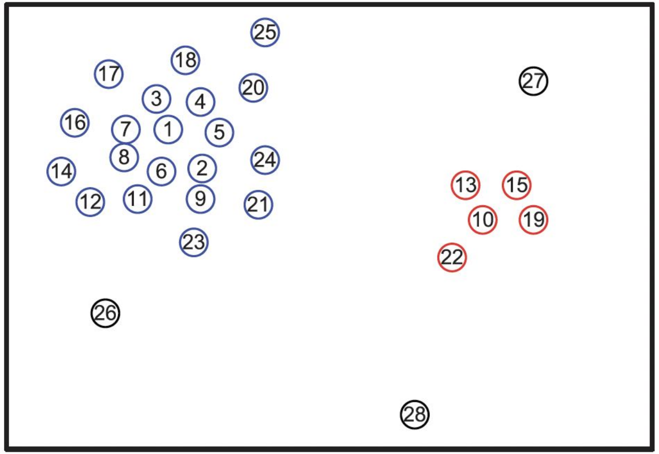
Density based Clustering



- Identify the cluster centers
- Then go to equation of ρ or δ to identify the points belonging to that cluster center

Density based Clustering: Algorithm

A



Step 5 — (Optional) Identify Outliers

Points with:

- Low ρ
- High δ

may be considered noise or halo points.

Clustering algorithms in High Energy Gamma Ray Experiment: FERMILAT

FermiLAT

- Fermi Large Area Telescope detects gamma rays with energy from 20 MeV to 300 GeV
- It investigates extreme phenomena like black holes, pulsars, supernova remnant, dark matter etc
- How does it detect gamma rays?
- It is a pair-conversion detector
  - Incoming gamma rays hit the tungsten sheet producing electron-positron pairs. Silicon strip detectors track thes particles, allowing the determination of the original direction of the gamma ray.
  - CeI calorimeter to measure the energy of these particle → Need clustering algorithms to reconstruct the energy

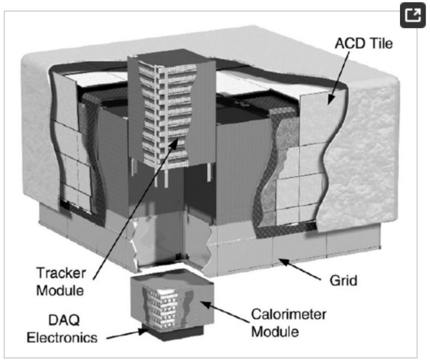


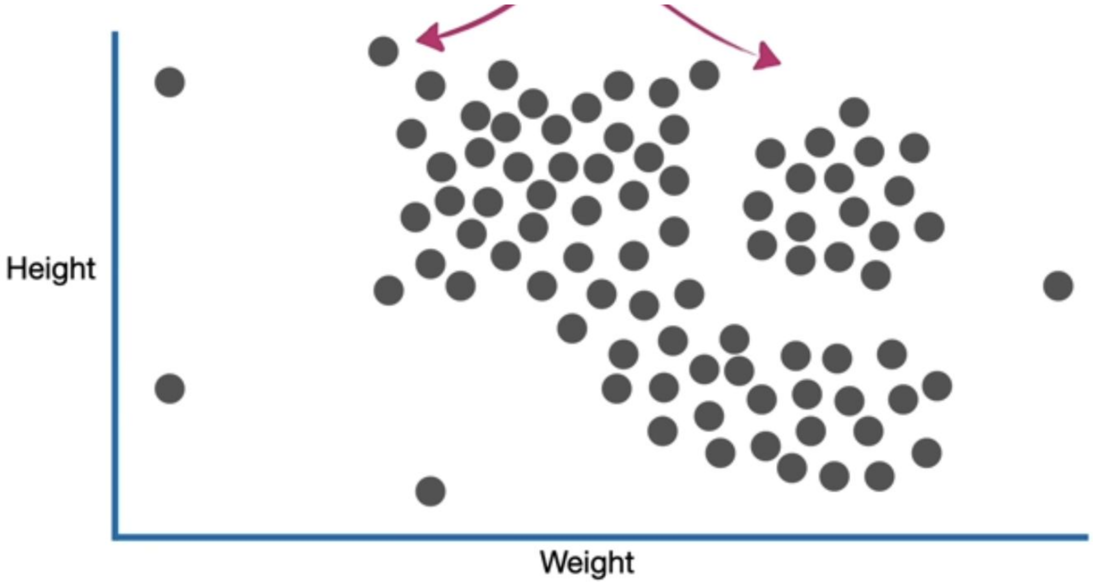
Figure 2. The Fermi Large Area Telescope (Fermi-LAT) instrument subsystems [14].

Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

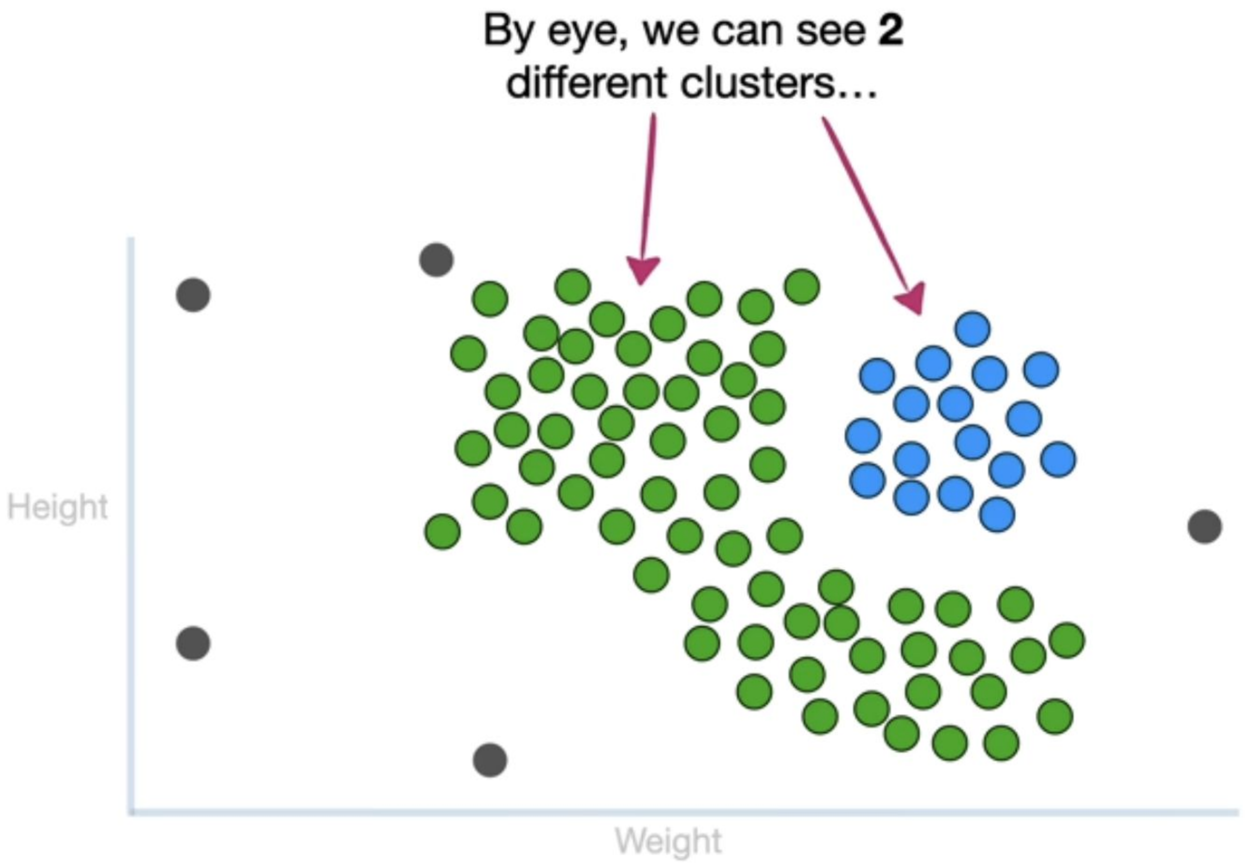
- FermiLAT uses Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to cluster the energy of gamma rays
    - Similar to what we learnt in the context of HGCAL

Next slides on the algorithm: StatQuest by Josh Starmer

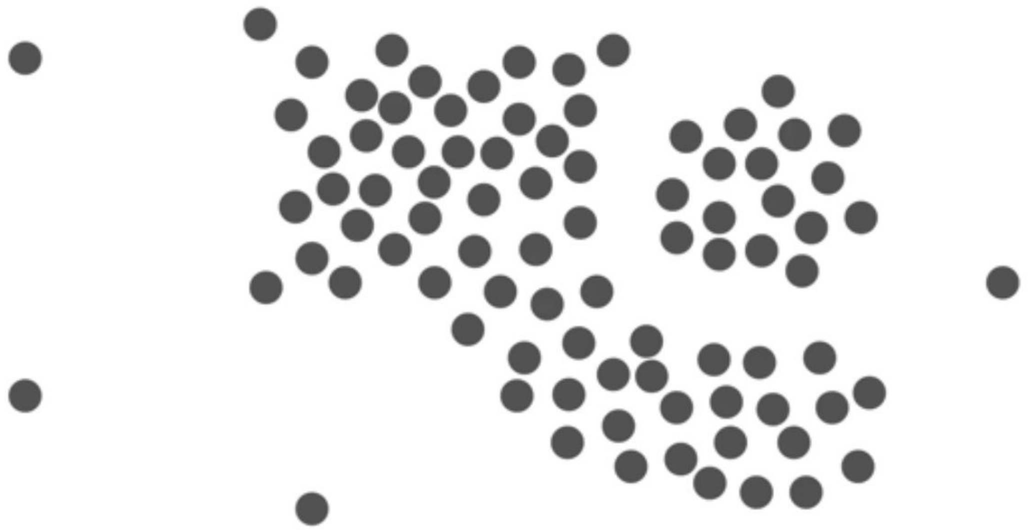
Let’s take an example



How many different clusters do you see?

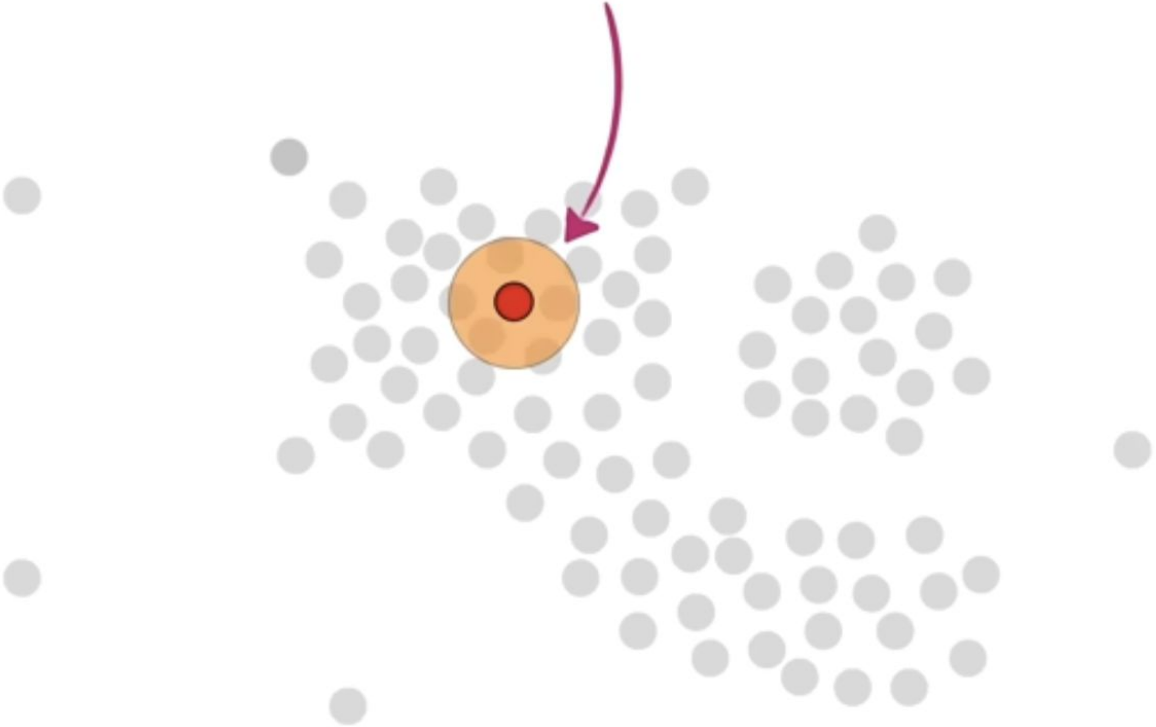

By eye, we can see **2** different clusters…
Height
Weight

…the first thing we can do is count the number of points close to each point.

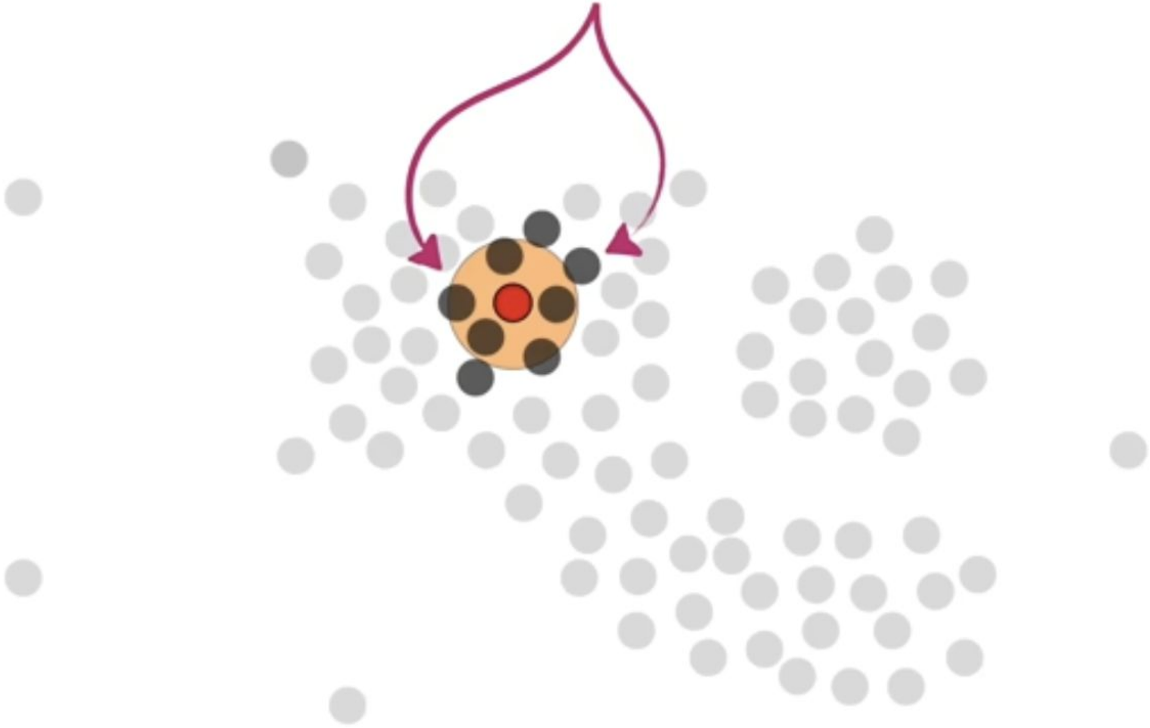


In this algorithm clusters are again identified by densities

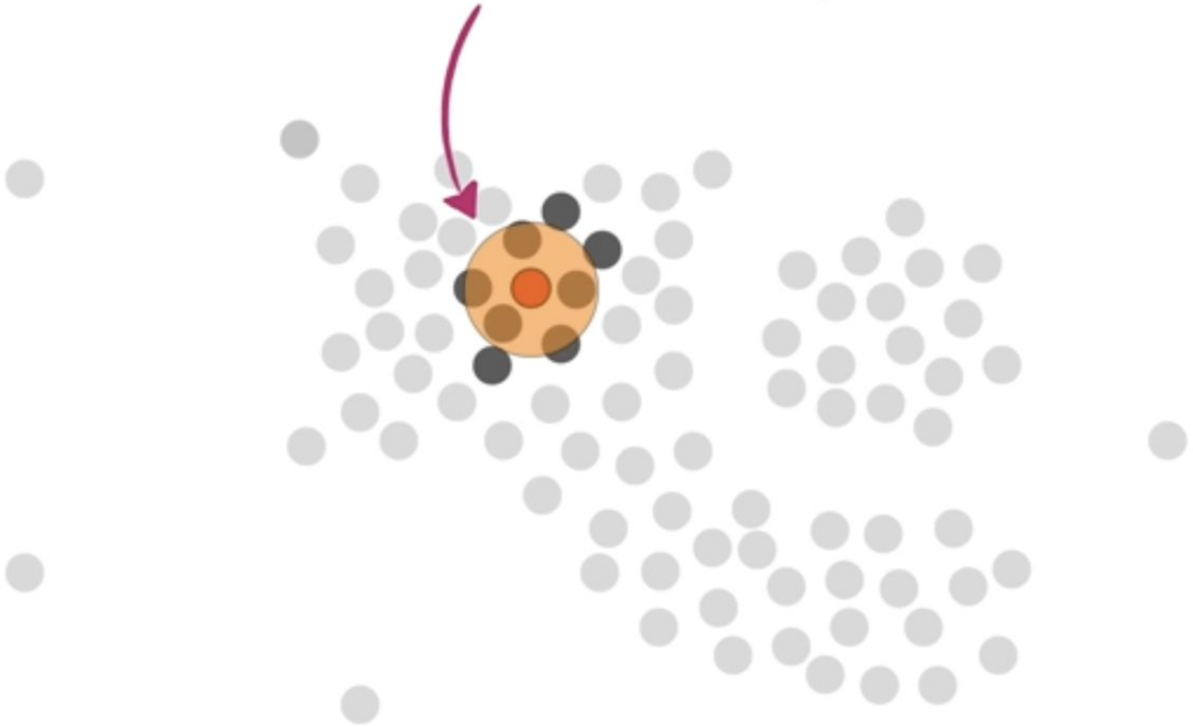
...and we draw an **orange circle** around it...



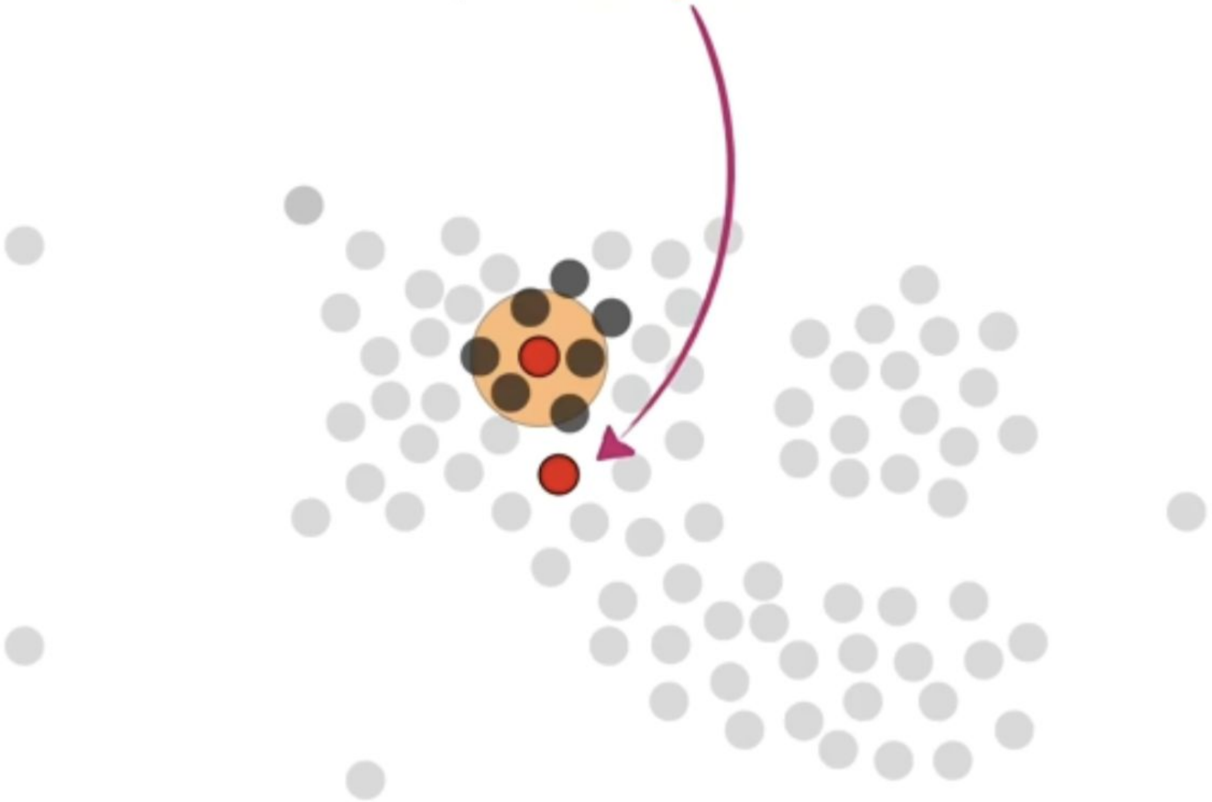
So the **red point** is
close to **8** other points.



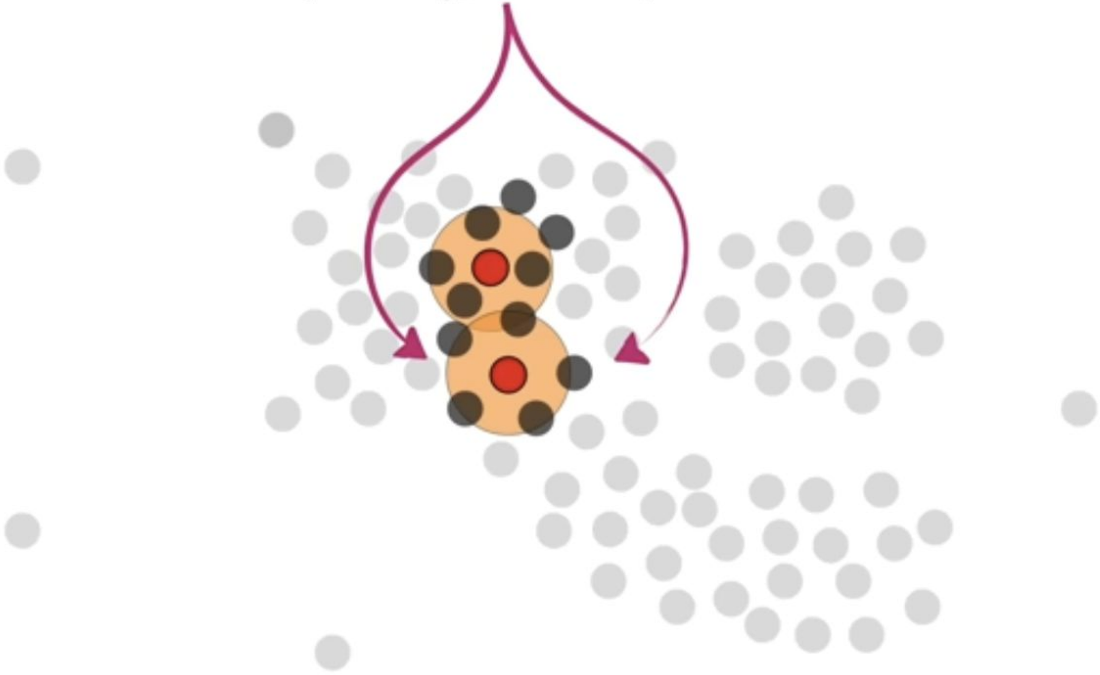
NOTE: The radius of the orange circle is user defined, so when using **DBSCAN**, you may need to fiddle around with this parameter.



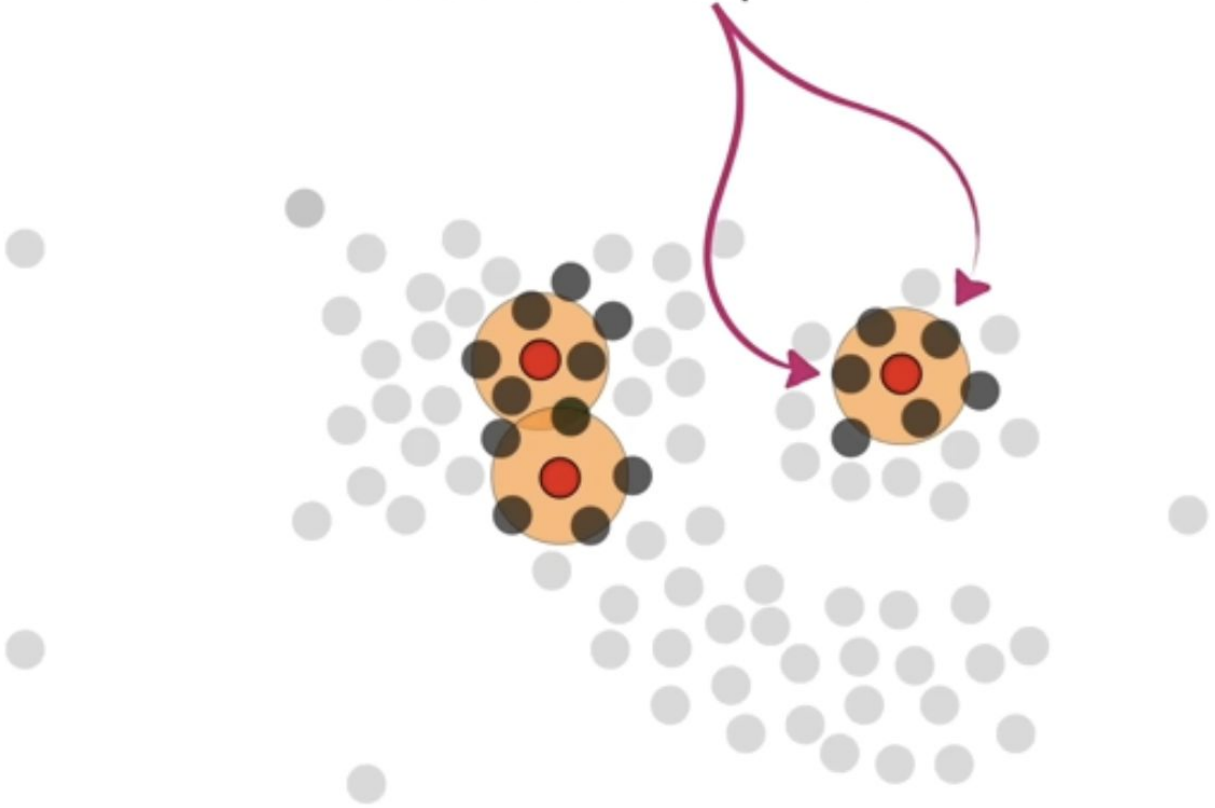
Now, this **red point**…

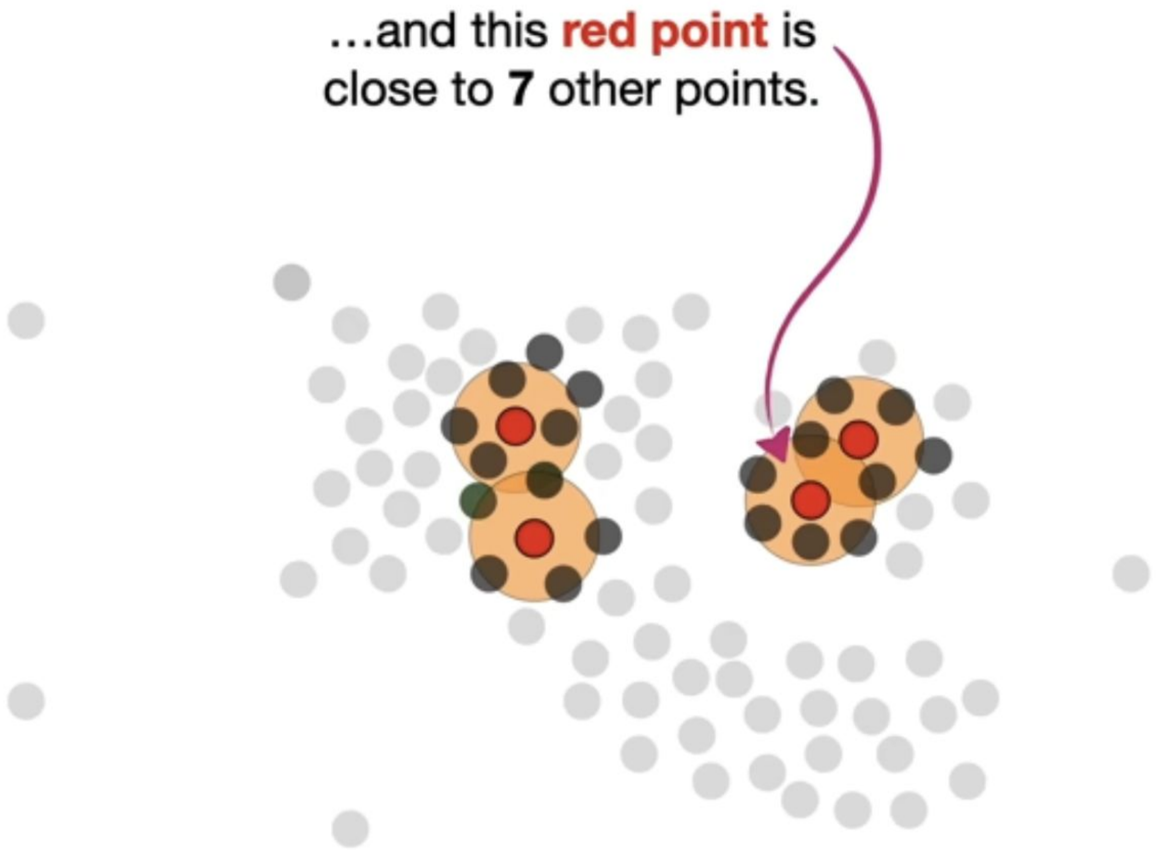


…is close to **5** other points because the orange circle overlaps, at least partially, **5** other points.

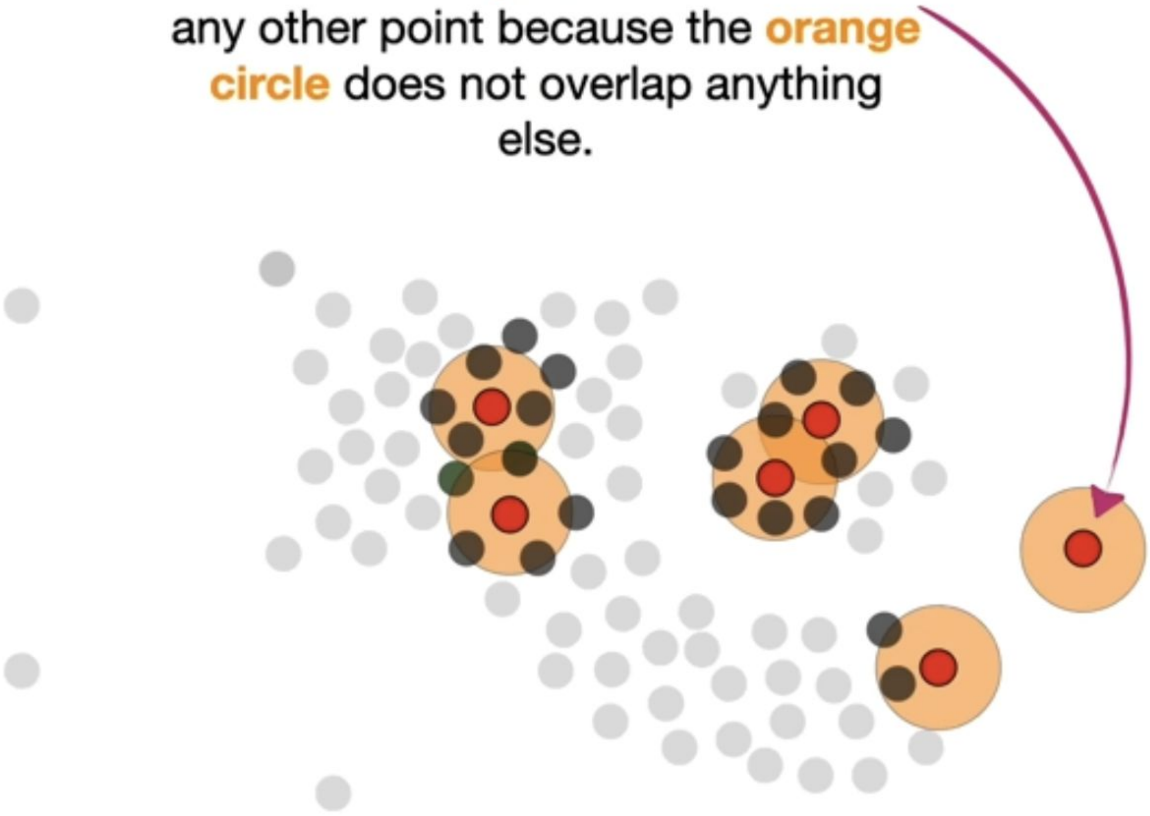


…is close to **6** other points…

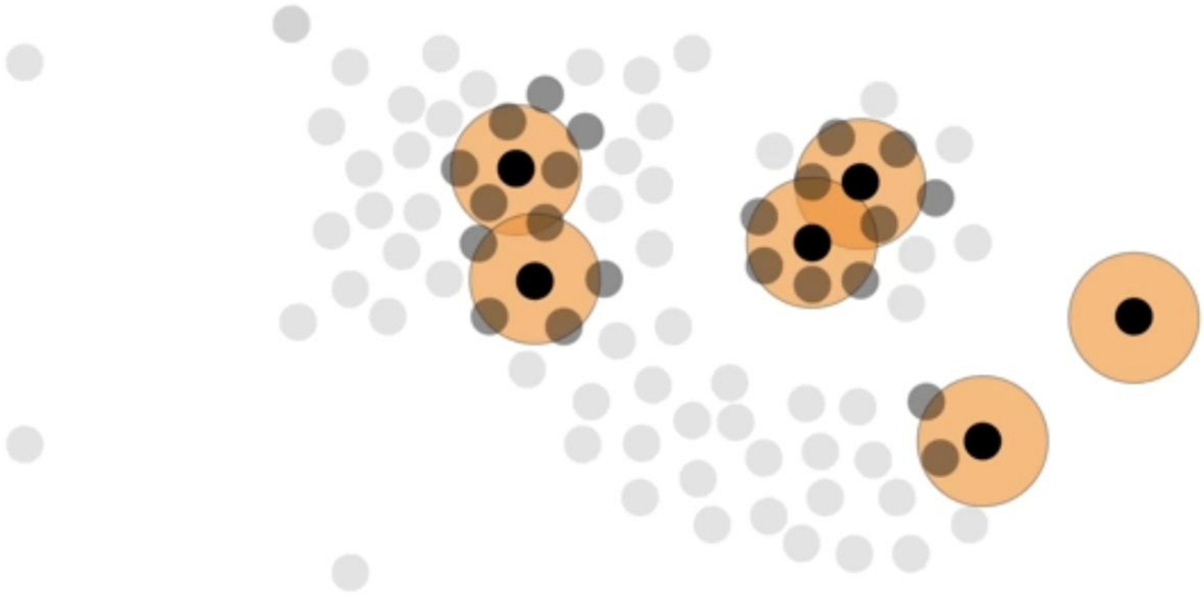



...and this **red point** is
close to **7** other points.

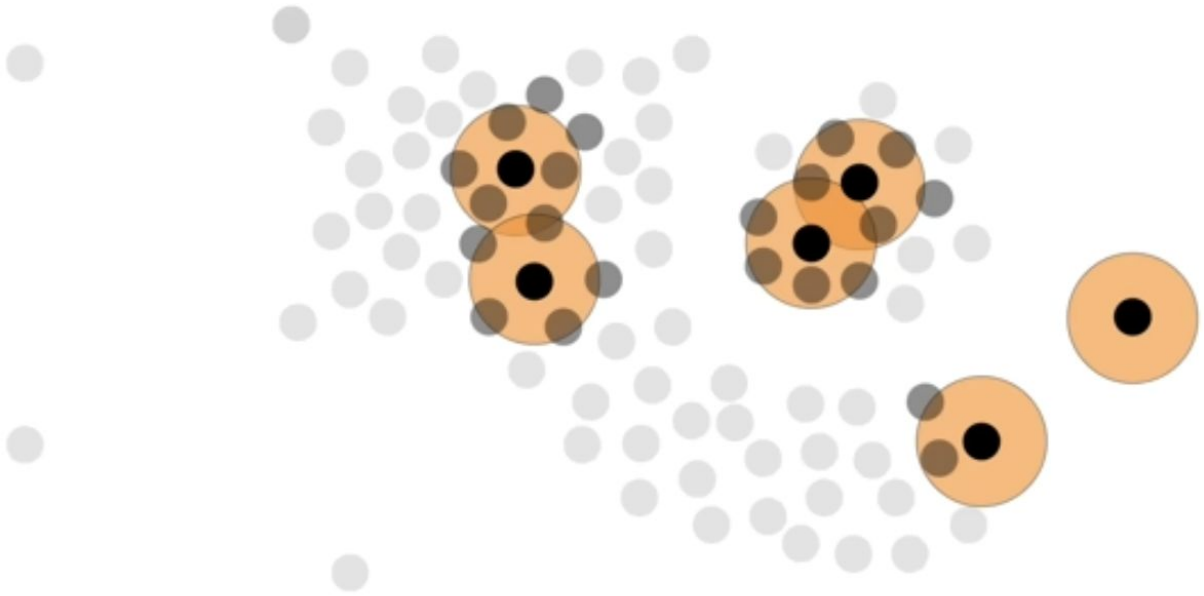
...and this **red point** is not close to any other point because the **orange circle** does not overlap anything else.



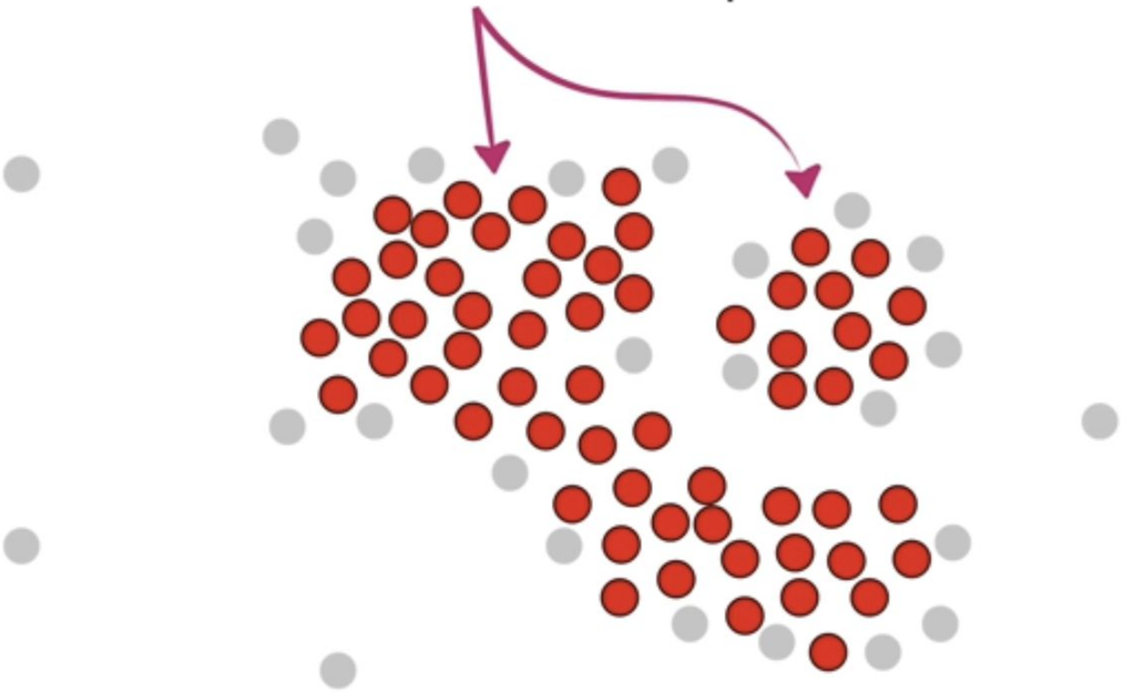
Likewise, for all of the remaining points, we counts the number of close points.



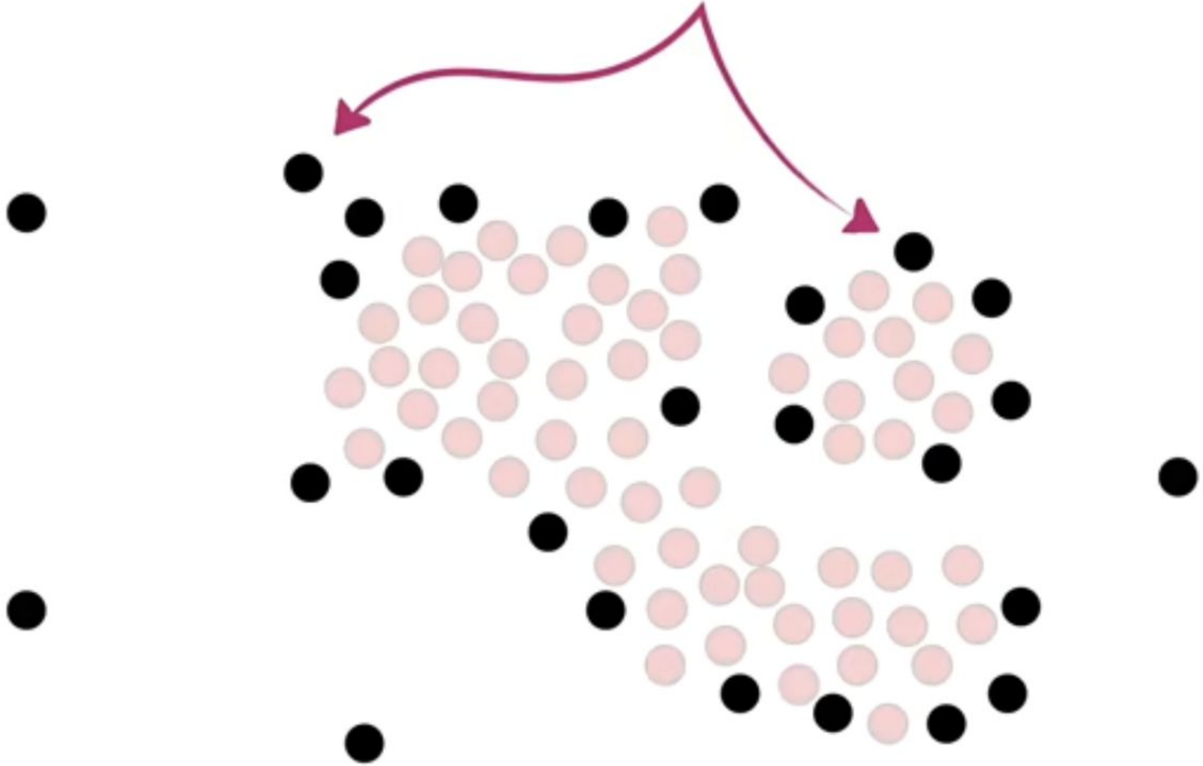
Now, in this example, we will define a **Core Point** to be one that is close to at least **4** other points.



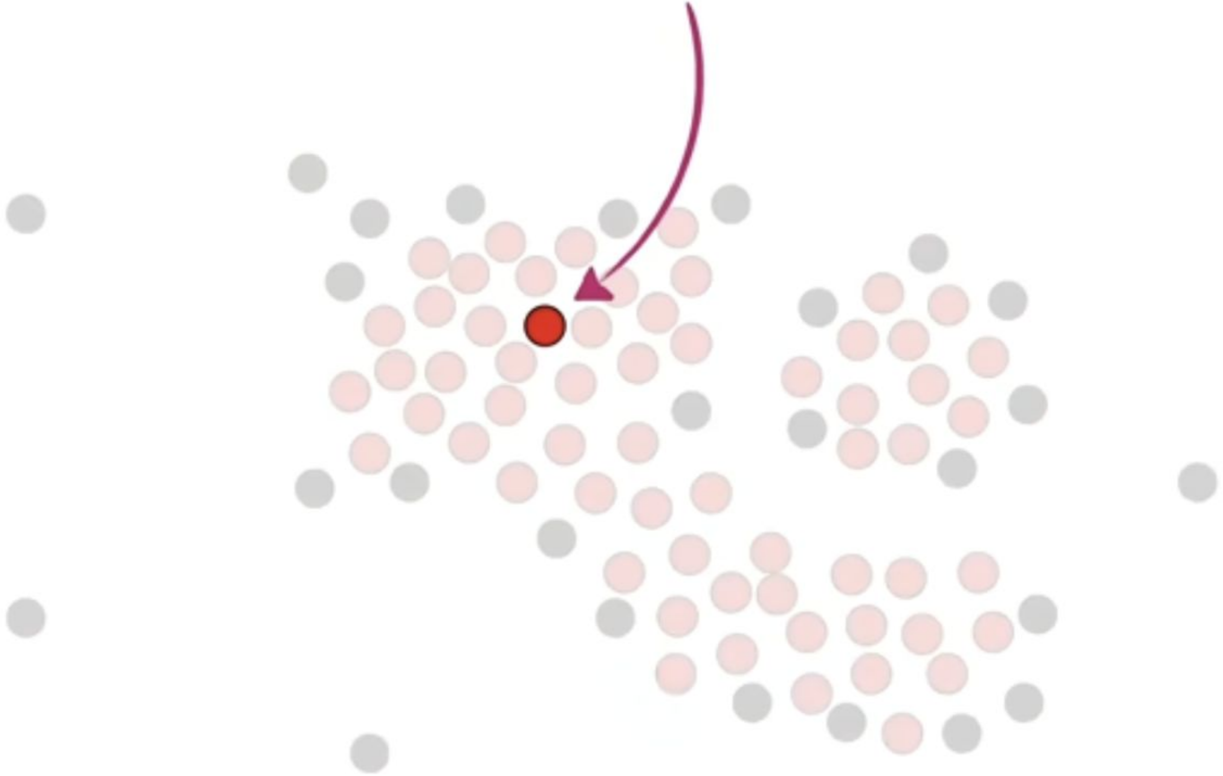
Ultimately, we can call all of these **red points** **Core Points** because they are all close to **4** or more other points…



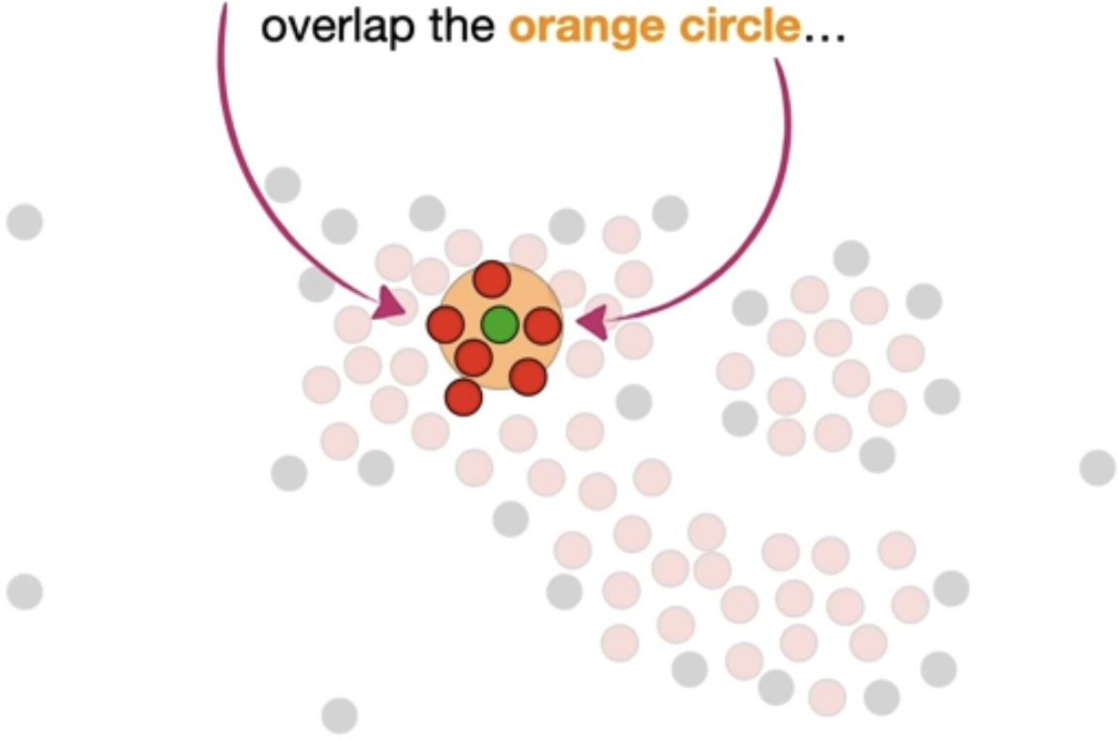
…and the remaining points are **Non-Core.**



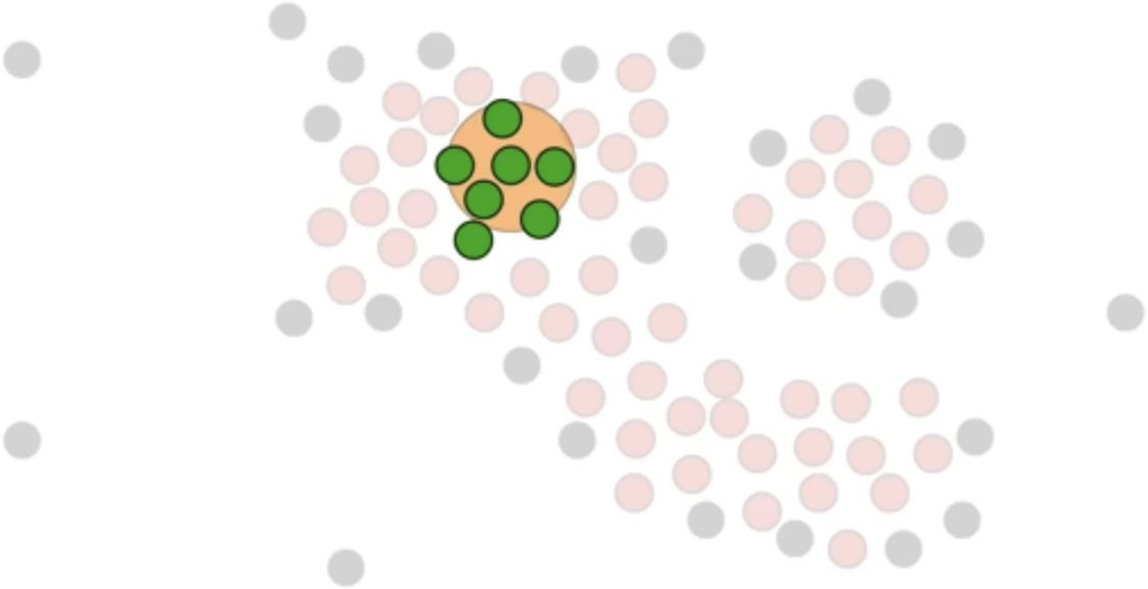
Now we randomly pick
a **Core Point**…



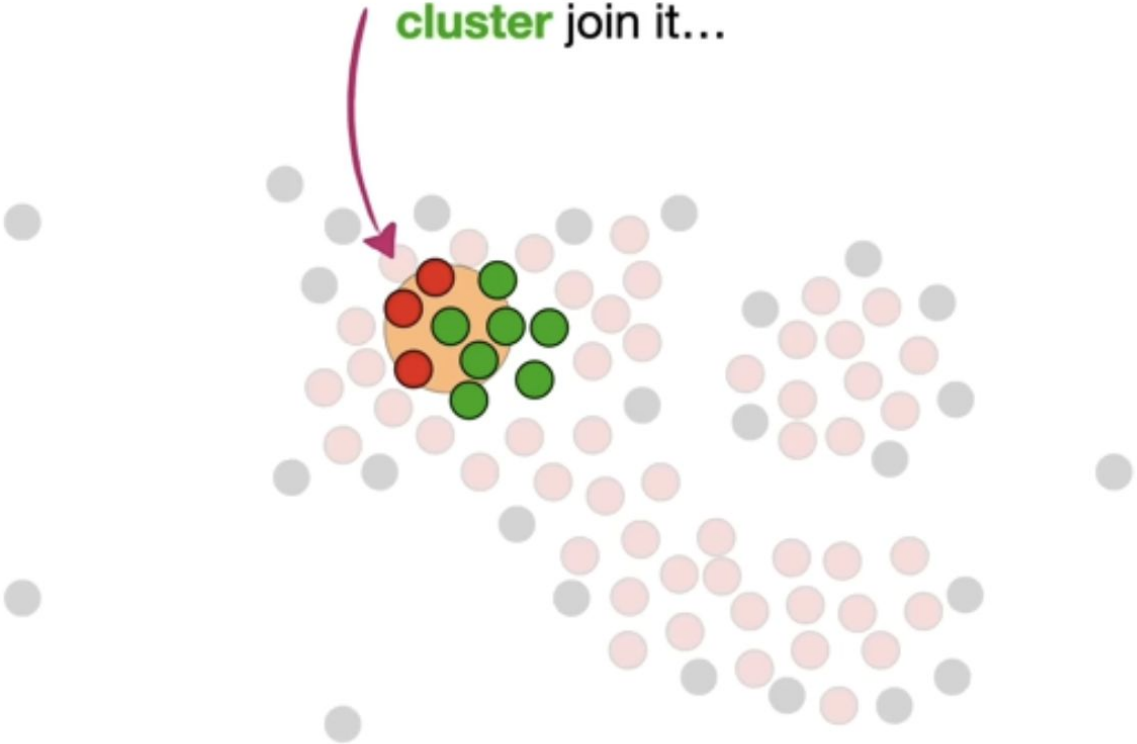
Next, the **Core Points** that are close to the **first cluster**, meaning they overlap the **orange circle**…

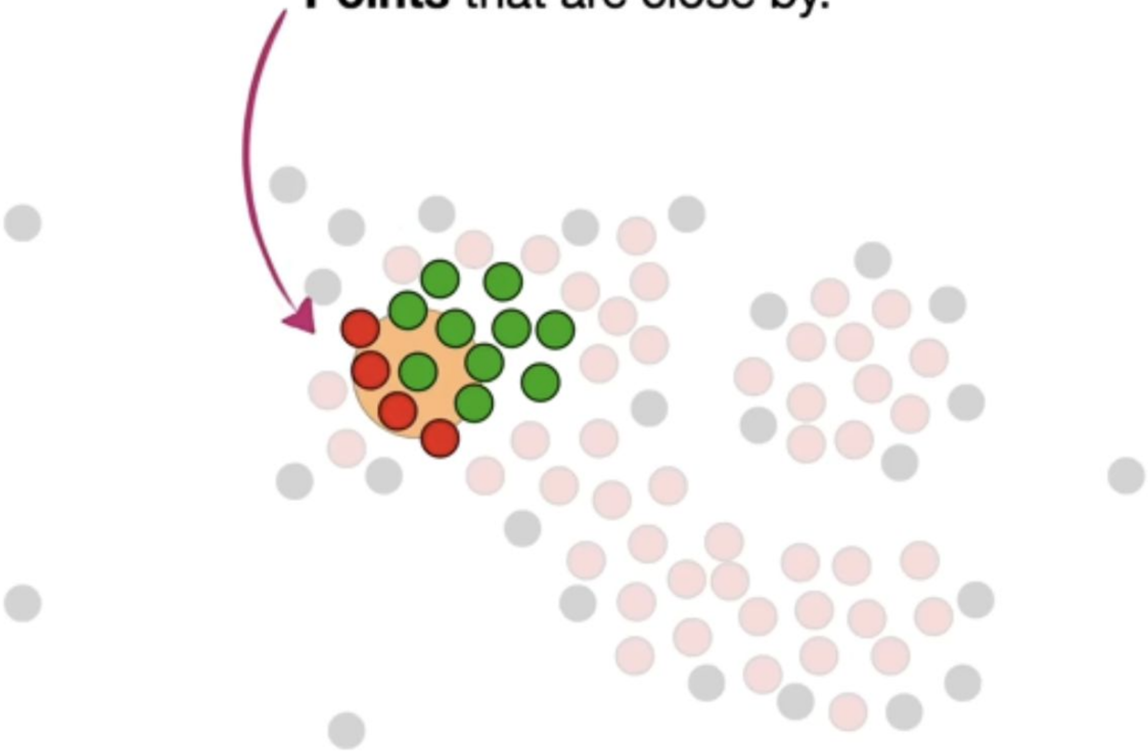


…are all added to the **first cluster**.

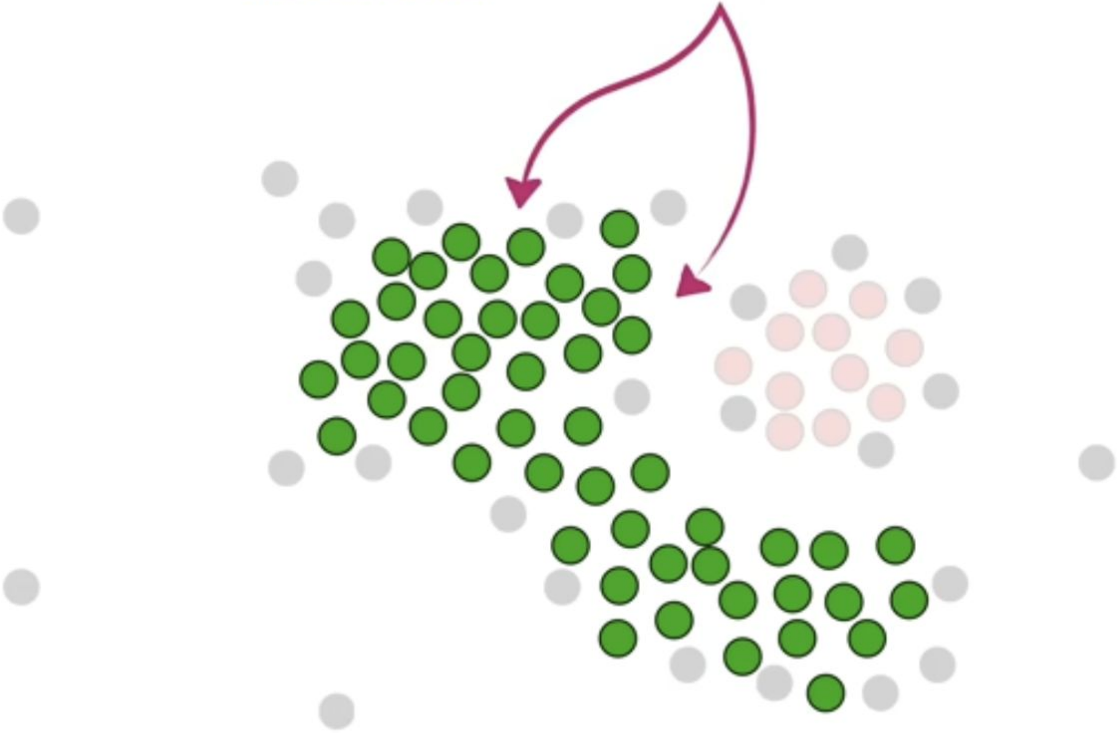


Then the **Core Points** that are close to the growing **first cluster** join it…

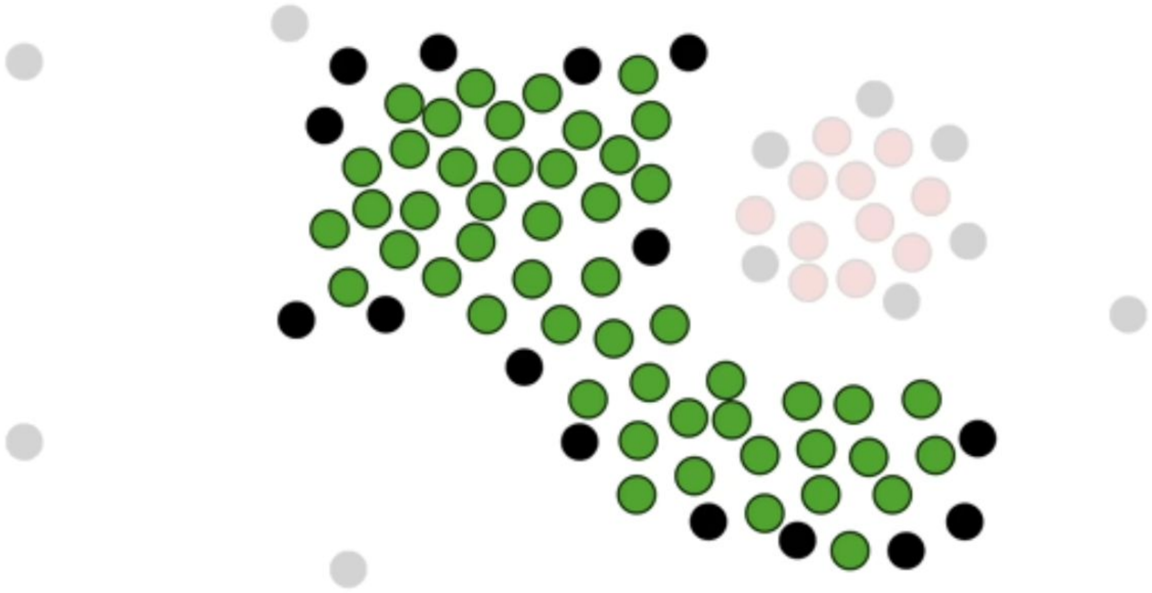



…and extend it to other **Core Points** that are close by.

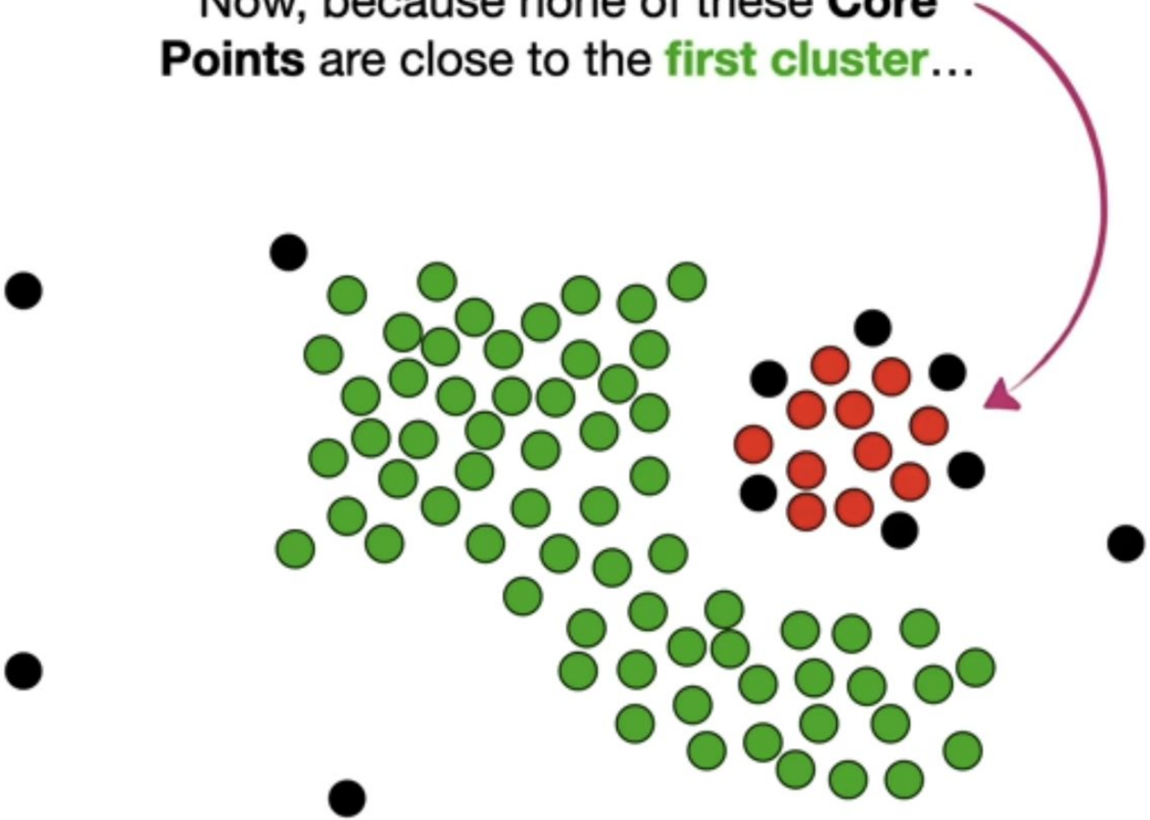
NOTE: At this point, every single point in the **first cluster** is a **Core Point**…

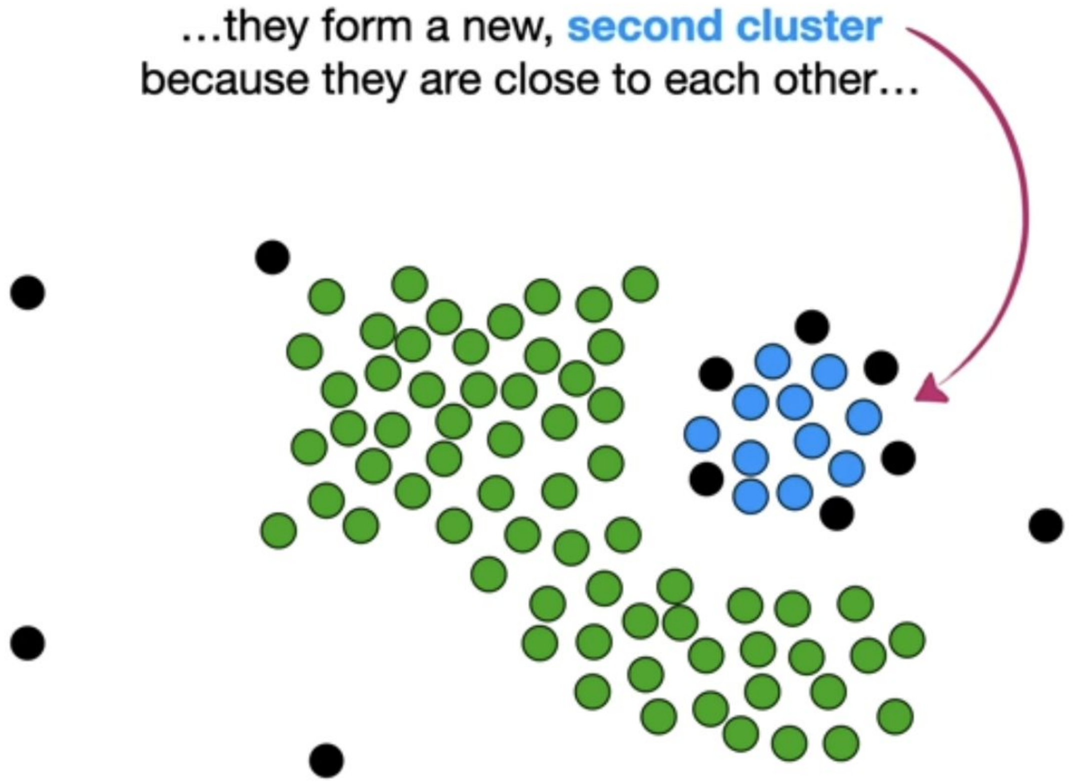


…we add all of the **Non-Core Points** that are close to **Core Points** in the **first cluster**.

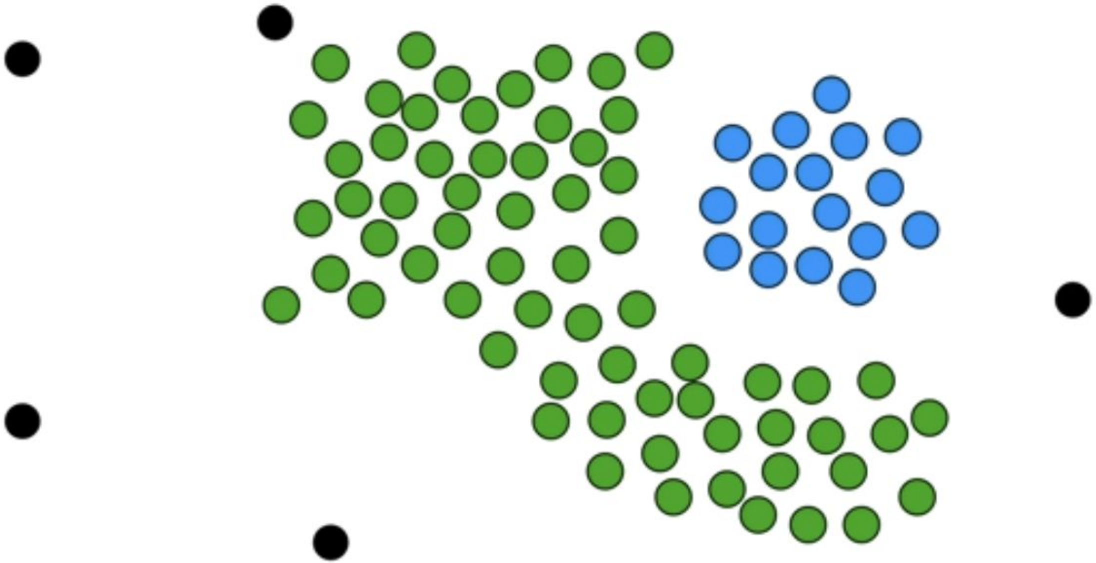


Now, because none of these **Core Points** are close to the **first cluster**…




…they form a new, second cluster
because they are close to each other…

Lastly, because all of **Core Points** have been assigned to a cluster, we're done making new clusters…



Clustering algorithms in Ice-Cube like experiments

Much of the material on generalities of IceCube are taken from Johannes Wagner's slides

Thanks to M. Rameez for the discussions

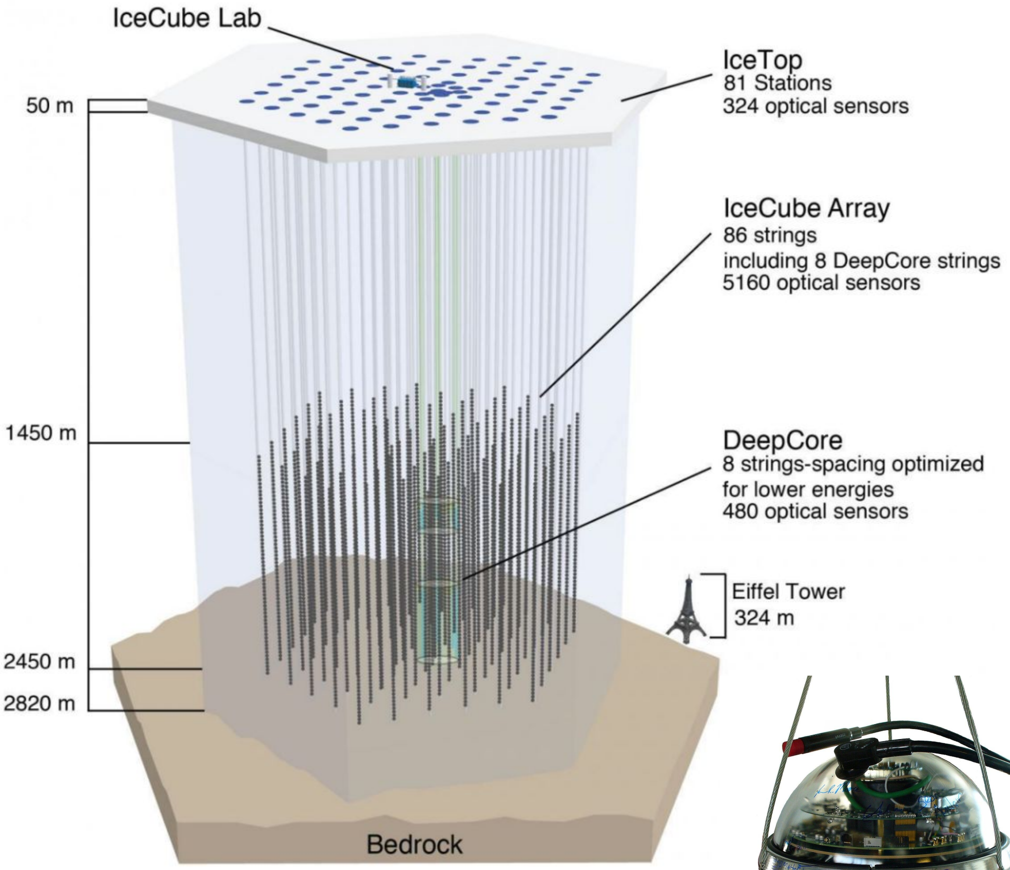
What is IceCube?

ICECUBE
SOUTH POLE NEUTRINO OBSERVATORY

- Centered around IceCube Neutrino Observatory
  - Cherenkov detector in Antarctica
  - Started data taking in 2008
- Focus on **neutrino astronomy**
  - General idea: observe the universe by looking at neutrinos rather than photons
  - Useful for observing distant phenomena since they scatter very rarely

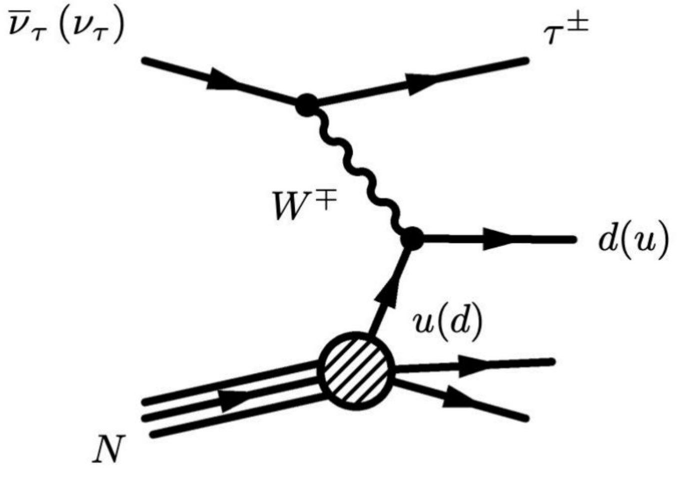
The IceCube detector

- Detector encompasses 1 km^3 of ice
- 86 strings with 5160 total Digital Optical Modules (DOMs) containing PMTs
- DeepCore: set of 8 central, densely-spaced strings

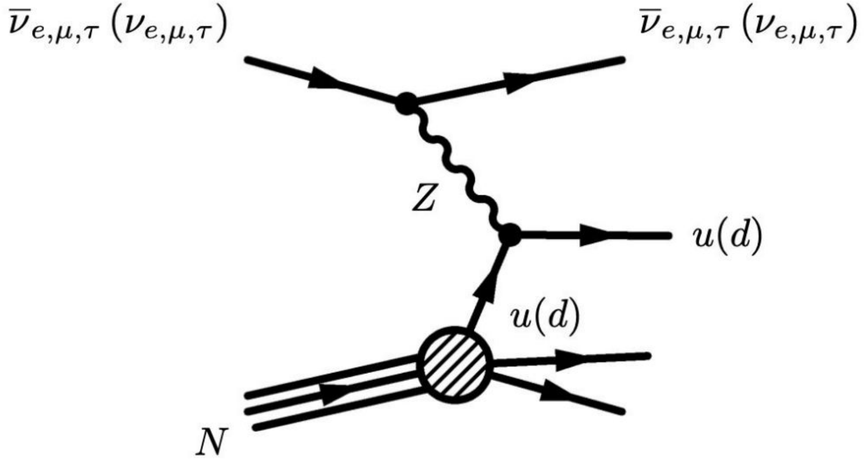


Experimental focus

- **High-energy neutrinos** (~TeV-PeV scale) created by violent interstellar events of particular interest for **astronomy**
  - Allow for identification of point sources in the sky (supernovae, gamma-ray bursts, black hole mergers, etc.)
- Can also detect **lower-energy neutrinos** created by cosmic ray interactions in the atmosphere (~GeV scale)
  - More interesting for **particle physics** purposes since travel distance is known (WIMPs, sterile neutrinos, oscillations, etc.)



Charged-current interaction



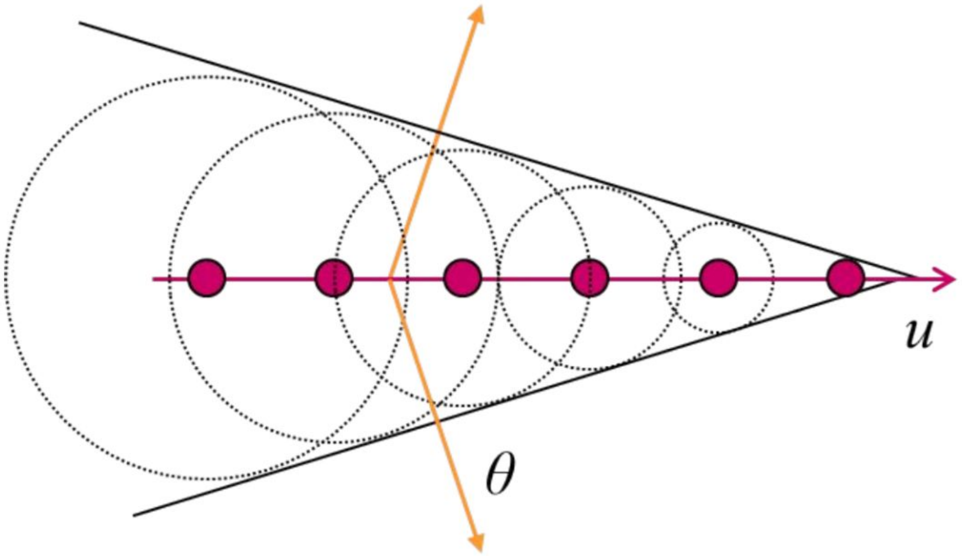
Neutral-current interaction

Cherenkov light is radiated by this primary lepton and any accompanying showers with a total amplitude proportional to the integrated path length of charged particles above the Cherenkov threshold.

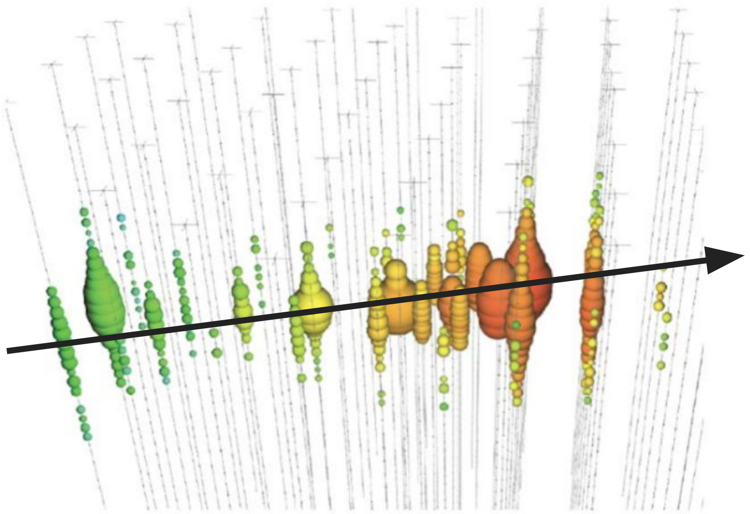
Similarly for the outgoing hadron

Cherenkov radiation

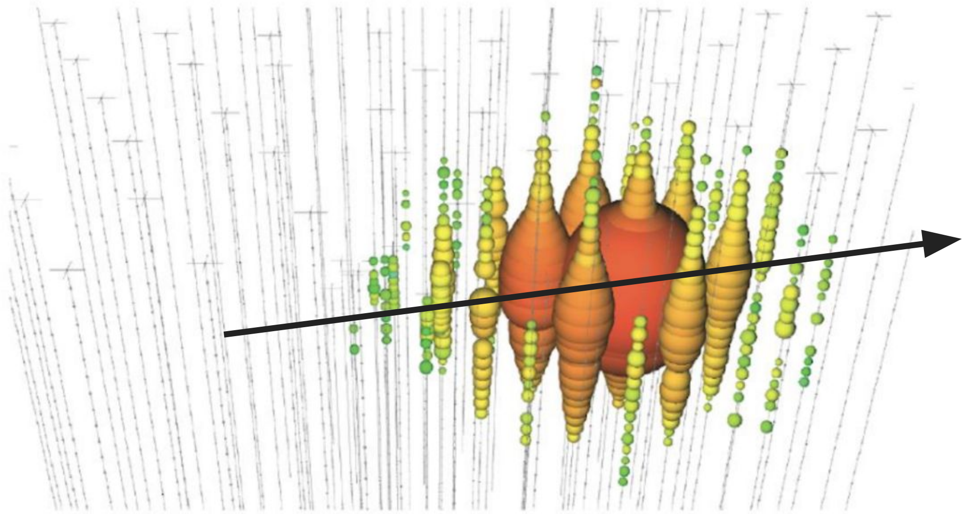
- Occurs when charged particles travel through medium at **v > c/n**
- Particle propagates faster than its own EM field -> interference
- Manifests as light cones along trajectory



Detector signatures



Tracks (produced by muons)



Cascades (produced by electrons, taus, hadronic showers)

Energy reconstruction using the likelihood model

- The light output scales linearly with energy
- From calibration, the number of photons from 1 GeV energy deposits are know, let that be Λ
- If k is the number of observed photons in a PMT, then the estimate a shower's energy deposition E can be estimated using the likelihood model

$$\mathcal{L} = \frac{\lambda^k}{k!} \cdot e^{-\lambda}$$

$$\lambda \to E\Lambda$$

$$= \frac{(E\Lambda)^k}{k!} \cdot e^{-E\Lambda}$$

$$\ln \mathcal{L} = k \ln (E\Lambda) - E\Lambda - \ln (k!) \,.$$

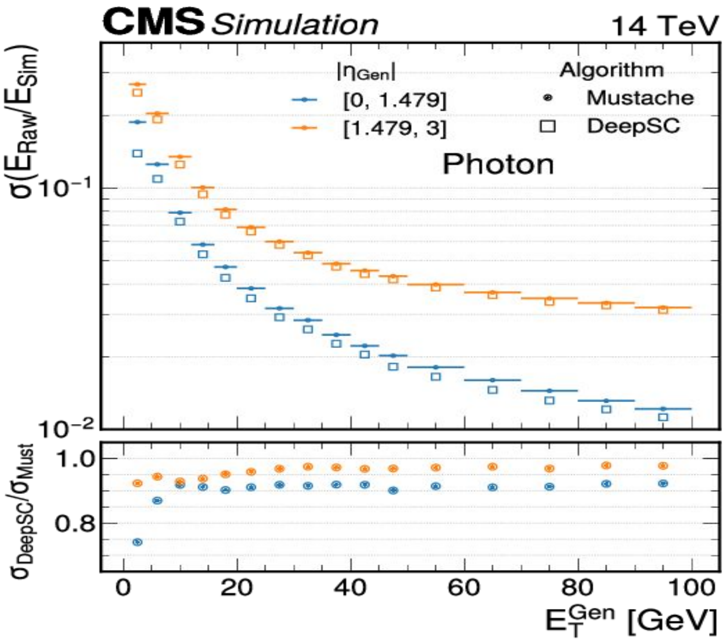
Energy reconstruction

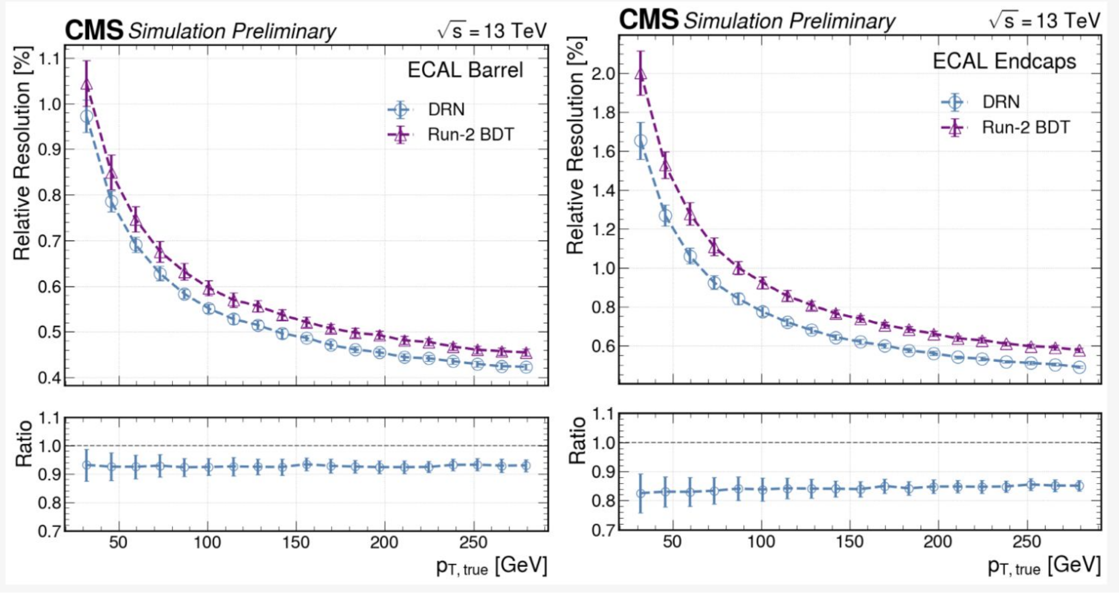
The number of detected photons is expected to follow a Poisson distribution with mean λ = ΛE. Then the likelihood L for an energy E resulting in k detected photons from an event producing Λ photons per unit energy can be evaluated as follows:

$$\begin{aligned}
\mathcal{L} &= \frac{\lambda^k}{k!} \cdot e^{-\lambda} \\
\lambda &\rightarrow E\Lambda \\
&= \frac{(E\Lambda)^k}{k!} \cdot e^{-E\Lambda} \\
\ln\mathcal{L} &= k\ln(E\Lambda) - E\Lambda - \ln(k!)\,.
\end{aligned}$$

$$\begin{aligned}
0 = \frac{\partial \sum \ln\mathcal{L}}{\partial E} &= \sum\nolimits_{\mathrm{DOMs}\ j} (k_j\Lambda_j / E\Lambda_j - \Lambda_j) \\
&= \sum k_j / E - \sum \Lambda_j \\
\therefore E &= \sum k_j / \sum \Lambda_j .
\end{aligned}$$

Emerging trends in clustering

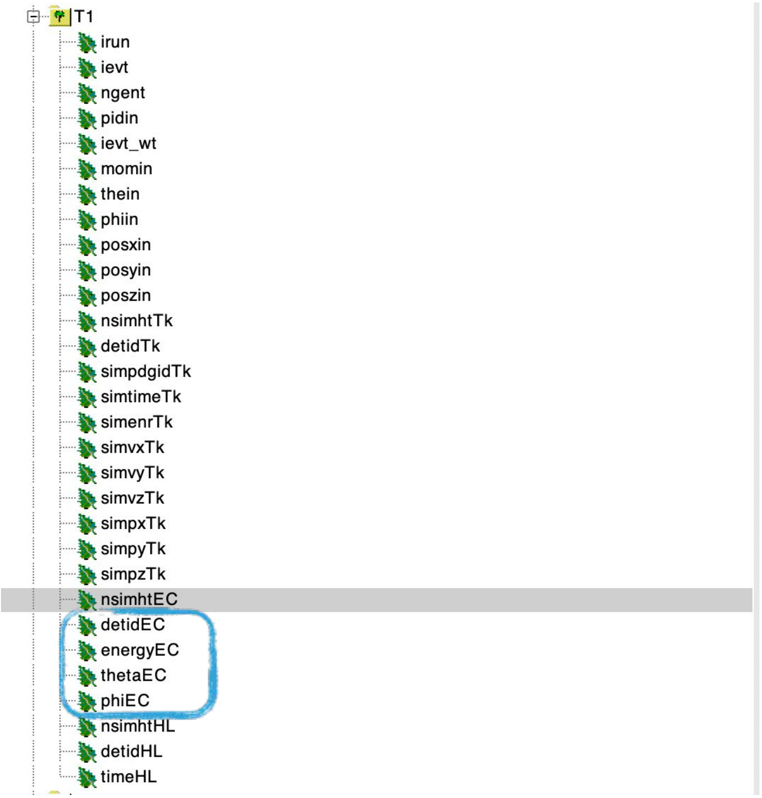




- Deep Neural networks, Graph neural networks etc have brought immense improvement in the energy resolution of the particles

EXERCISES

ROOTFILE



- Code repository:
  - git clone https://github.com/jainshilpi/EHEP2026.git
  - Ex 1 and 2: serc19_ecal_clustering.C
  - Ex 3(a): myEMalgo.C
  - Ex 3(b) and remaining (time permitted): serc19_ecal_clustering.C
  - Use Makefile for compilation

ROOTFILE

Already in the file

```
for (int ij = 0;  ij<nhit; ij++) {
  int ienr = (detid[ij]&0x3ffff);     //Digienergy in MeV
  //One  can add noise here to, but that will be biased due to already ped suppression
  double enr = ienr/1000.0; //convert to GeV

  simenr += enr;
  if (enr <thresh) continue;
  digienr +=enr;

  int ithe = ((detid[ij]>>25)&0x7f);  //itheta during coding
  int iphi = ((detid[ij]>>18)&0x7f);  //iphi during coding

  double the = (ithe + 45.5)*degtorad;
  double phi = (iphi - 44.5)*degtorad;

  tmp3vect.setRThetaPhi(enr, the, phi);

  hitpos.push_back(tmp3vect);
```

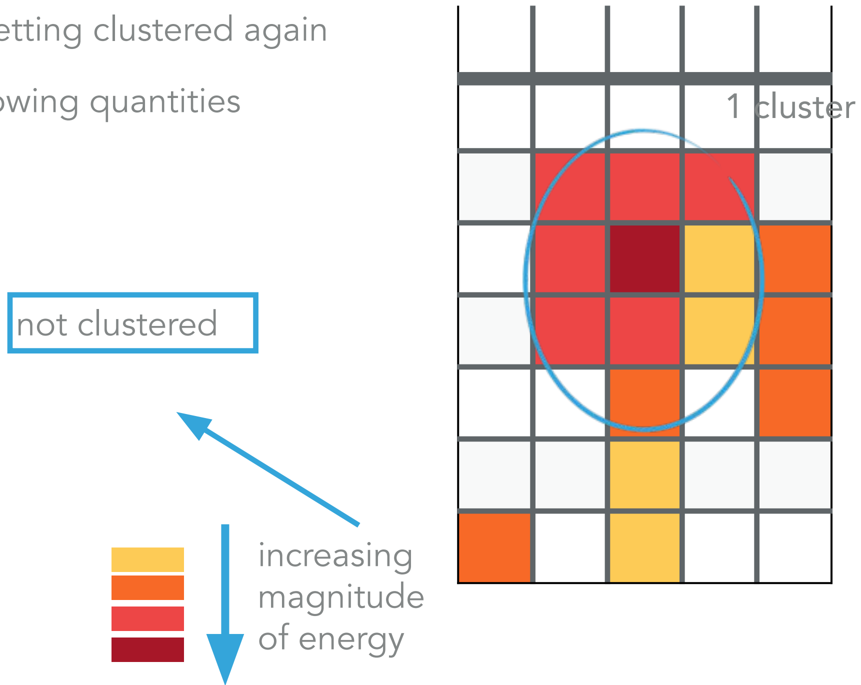
- How to use the above information to do the exercises?

```
for (int ij=0; ij<digienr.size(); ij++) {
double energyhit = digienr[ij].mag();
double cell_theta = digienr[ij].theta()*radtodeg;
double cell_phi   = digienr[ij].phi()*radtodeg;
}

To calculate angle between two crystals:
double angle = cluster[jk].angle(digienr[kl].unit());
```

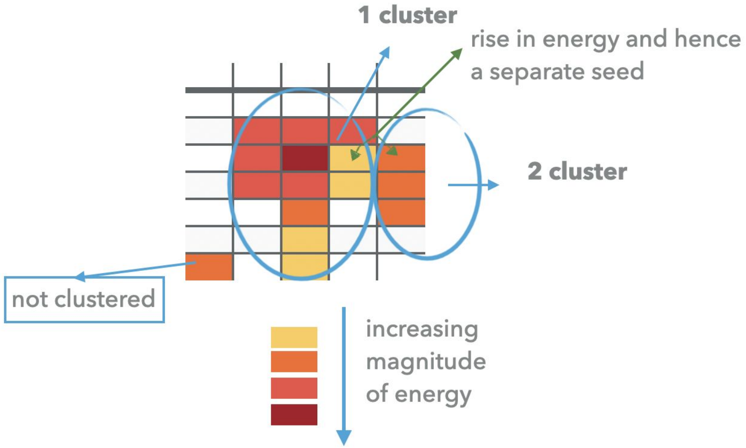
EXERCISE 1

- Cluster the hits from a photon using basic clustering algorithm:
  - Make a collection of crystals passing seed threshold criteria. Take seed threshold energy = 0.5 GeV
  - Start with this collection, and collect crystals for each seed which are within 3x3 vicinity around it (passing a noise threshold of 0.04 GeV to start with) or in the vicinity of already clustered crystal.
    - Remove the hits which are being clustered from getting clustered again
- Associate each cluster with a photon and form the following quantities
  - E9, E25
  - Form moments:
    - dTheta = Σ(theta_hit - theta_seed)*energy_hit
    - Similar for dPhi and dPhidTheta
    - Diagonalize the matrix



EXERCISE 2

- Add the condition in exercise 1 that while clustering there is no rise in energy as we go in one direction.
- If there is a rise, then take that hit as another seed
- Again compare as in Ex1



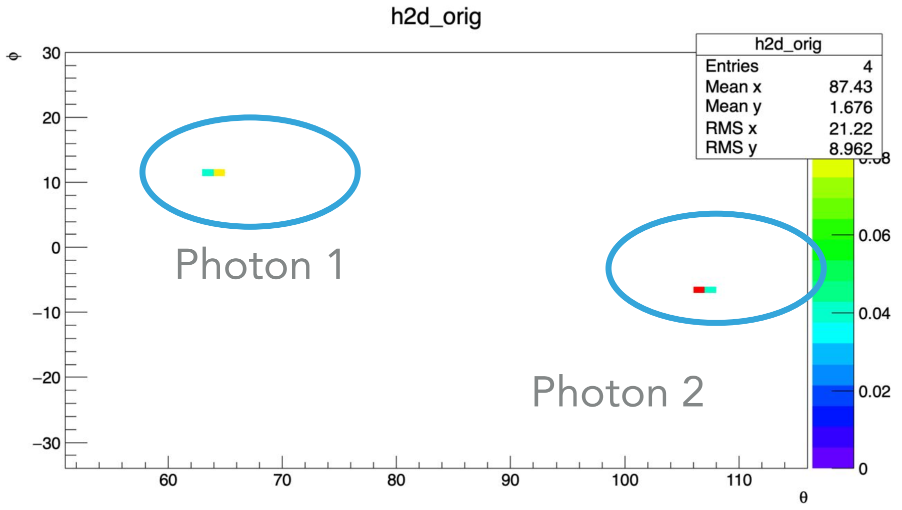
EXERCISE 3(A)

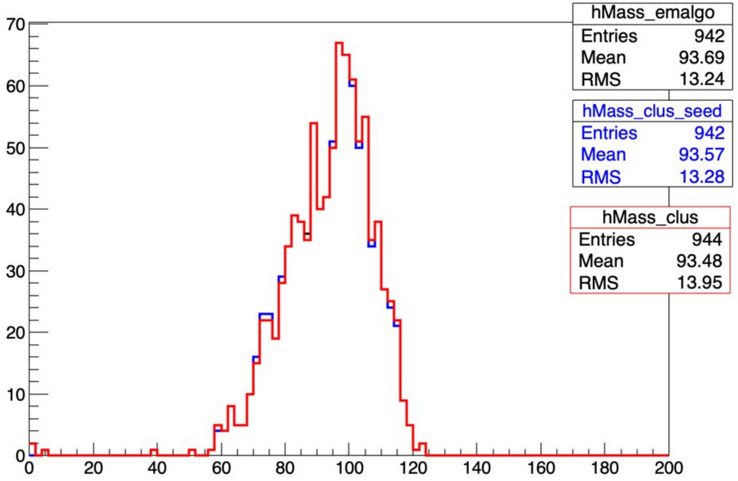
- Use EM algorithm
  - Start with 1-D pseudo data
  - You will see 3 gaussians with known mean and sigma are filled in a histogram.
  - Now consider each bin of the histogram as a crystal and apply EM algorithm
  - Check if you can get similar means and sigma of the gaussian as you started with

EXERCISE 3(B)

- Now that you have learnt how to use EM algorithm in1-D, use EM algorithm on the hits and extend to 2D EM algorithm
- Compare this with the algorithm in exercise 2 by forming π^0 mass

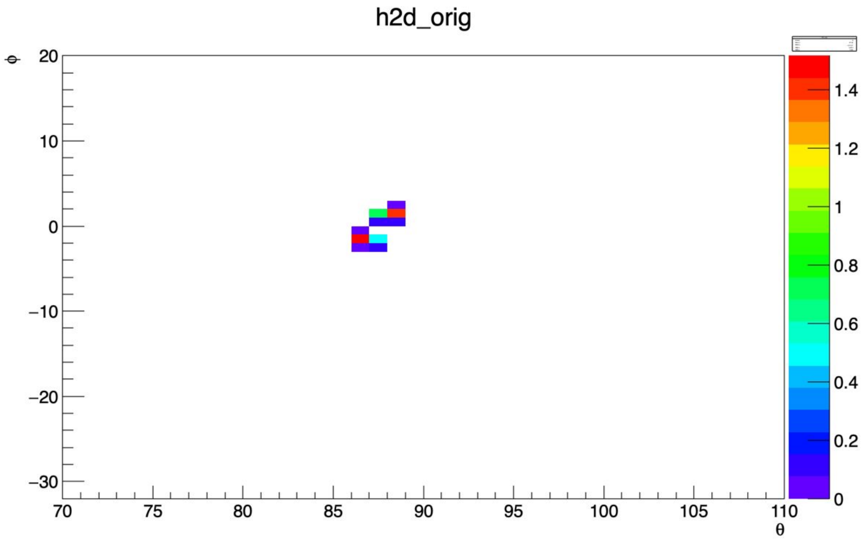
Π0 mass : 150 MeV energy pions

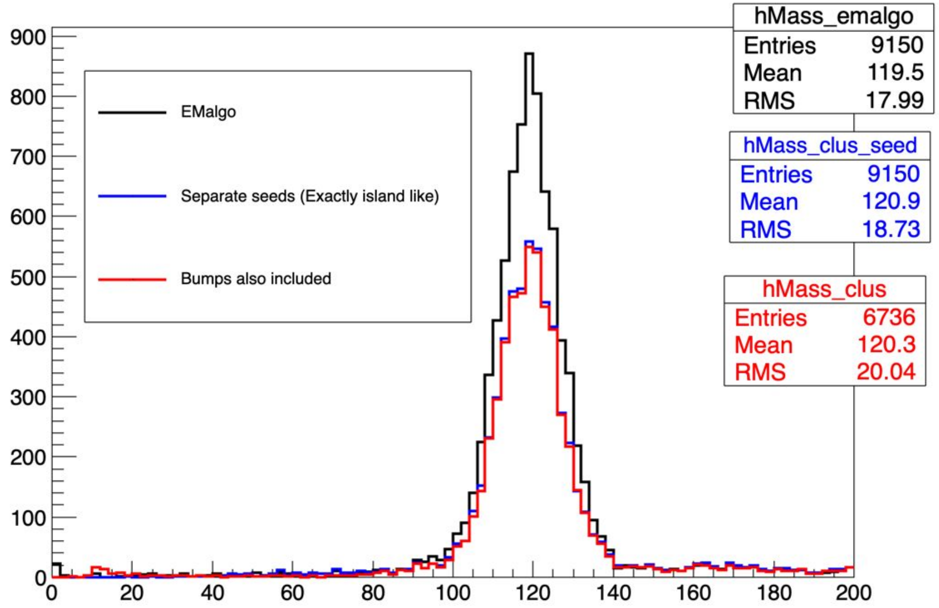




150 MeV pions: Two clusters are very separated so the performance between EM algorithm, high energy bump separated algorithm and the default (Where hit with high energy is also included if falls in the way of a neighbouring hit)

Π0 mass : 5 GeV energy pions





5 GeV pions: Showers are very close and hte performance between different algorithms is different

EXERCISE 2

- Use the quantities thus formed to determine the noise thresholds

EXERCISE 4

- Energy regression using the photon variables

EXERCISE 5

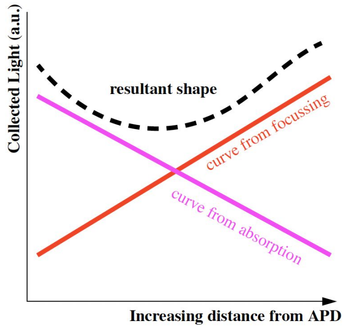
- Discrimination between pi0 and gamma using
  - Genetic algorithm

BACKUP

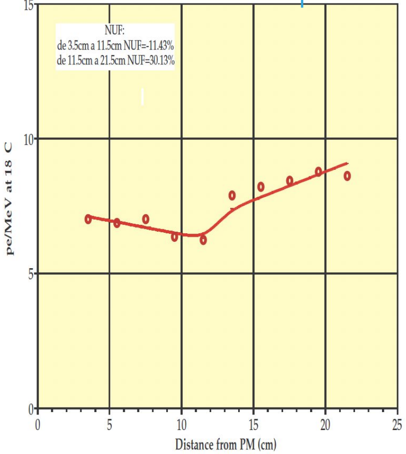
Detecting neutrinos

- Neutrinos only interact weakly -> **cannot measure directly**
  - Need to examine products of their interactions
- IceCube detects **Cherenkov radiation** (photons) originating from particles produced in neutrino interactions via its array of PMTs
  - Incoming neutrinos interact with protons/neutrons in the ice
- Antarctic ice allows photons to travel relatively undisturbed (minimal scattering)

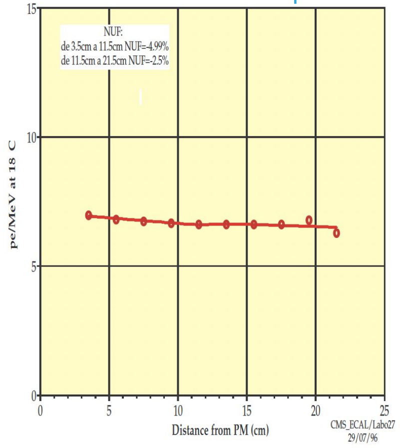
LIGHT COLLECTION EFFICIENCY

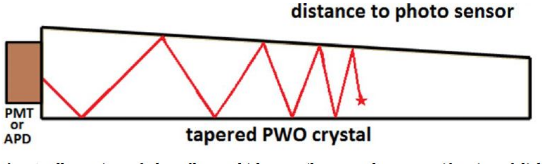


Before depolishing



After depolishing



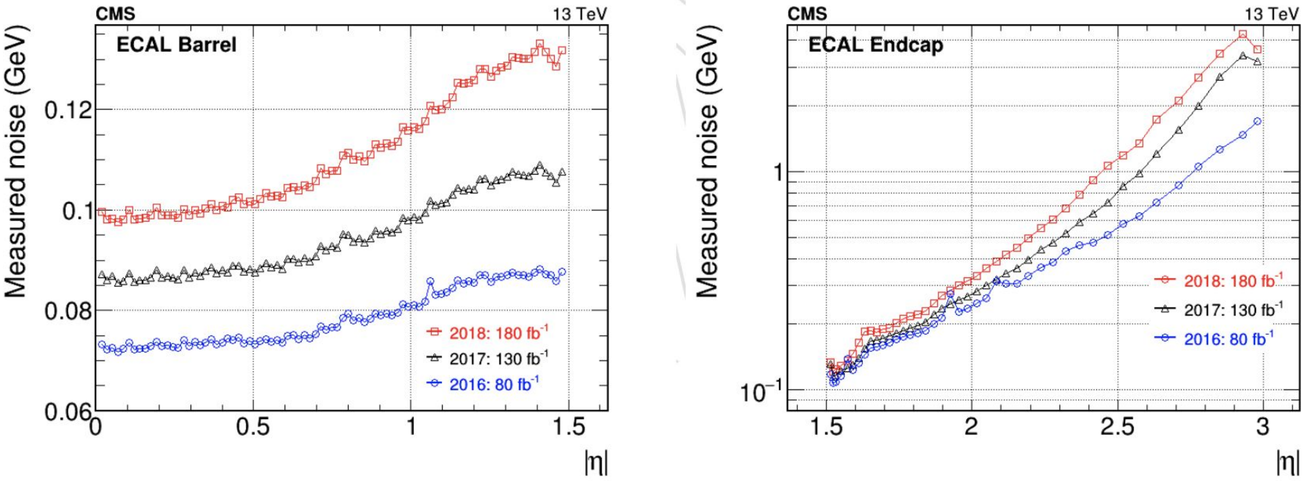


Depiction of focusing effect

- The crystals that went inside the CMS detector had increased absorption lengths —> light does not get absorbed so fast as shown in the left plot above
  - Resulting light collection curves are dominated by focusing effects
  - This could introduce non-uniformity in the light collection and hence increase the constant term in the resolution
  - One side was depolished to reduce the focusing effect

References:
- Light collection uniformity of lead tungstate crystals for the CMS ECAL: NIM A 540(2005) 273-284
- NIM A 857 (2017) 1-6
- Internal CMS note: CMS CR 1998/004

Noise level in the ECAL Barrel and the Endcap



(c) EB, *E*

(d) EE, *E*

Noise increase in the EB due to increase in APD dark current

Noise increase in the EE due to decrease in transparency and

[CMS-DP-2022/015]

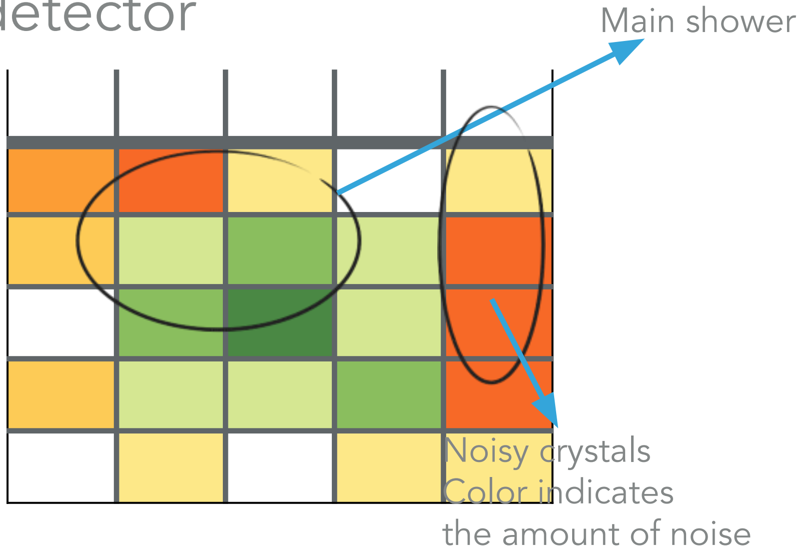
Effects of noise on physics

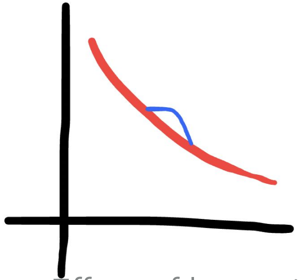
- Noise worsens the resolution of the detector

$$\frac{\sigma(E)}{E} = \frac{a}{\sqrt{E}} \oplus \frac{b}{E} \oplus c$$

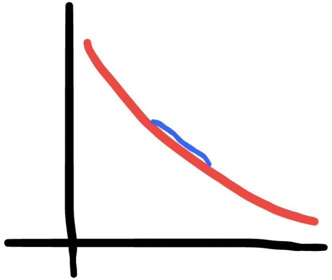
Noise term

  - Such noisy crystals are picked up in the clustering algorithm and worsen the total resolution of electron and photon object





Effect of low noise on the signal peak sitting on the background continuum



Effect of high noise on the signal peak sitting on the background continuum

  - Remove noise by applying a threshold
  - Threshold tuned in such a way that resolution is affected the least

EE TRANSVERSE FACE DIMENSIONS

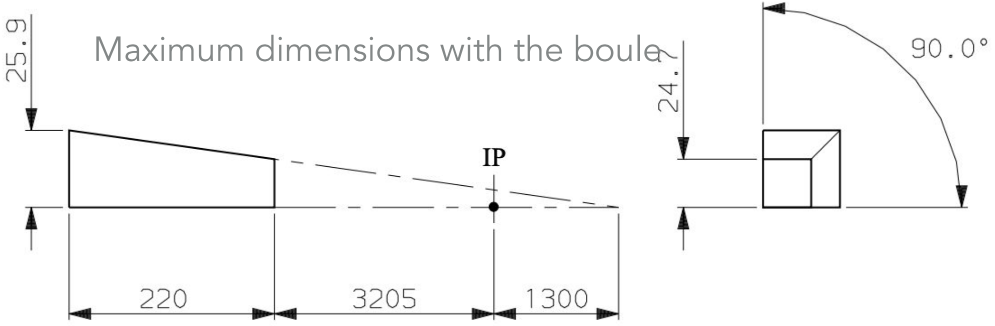


Fig. 3.28: EE crystal geometry (not to scale, dimensions in mm).

The geometric construction of the EE is based on a right-sided crystal with two tapering sides as shown in Fig. 3.28. The taper is defined by a line from a point 1300 mm from the far side of the intersection point, to the rear corner of the crystal. The taper defines the size of the front face of the crystal. The maximum crystal width, at the rear, that can be obtained with the current crystal boules is 25.9 mm. The corresponding front-face width is 24.7 mm. The taper on the crystal is small, only 1.2 mm over the full crystal length of 220 mm. Off-pointing to the far side of the intersection point is required in order to ensure maximum path length through the EE crystals.

ECAL READOUT CHAIN

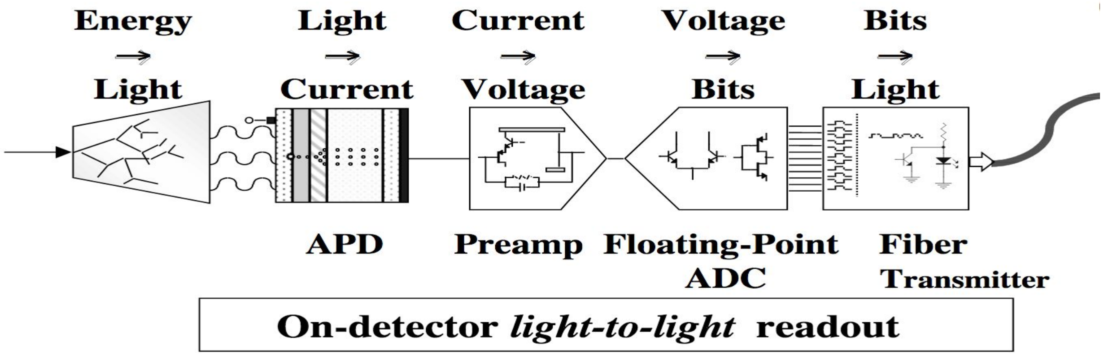
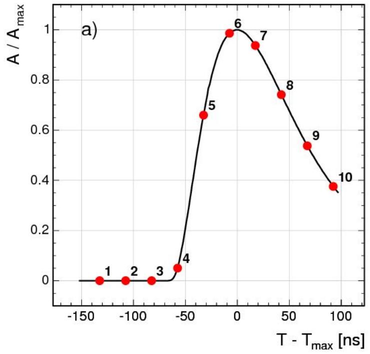
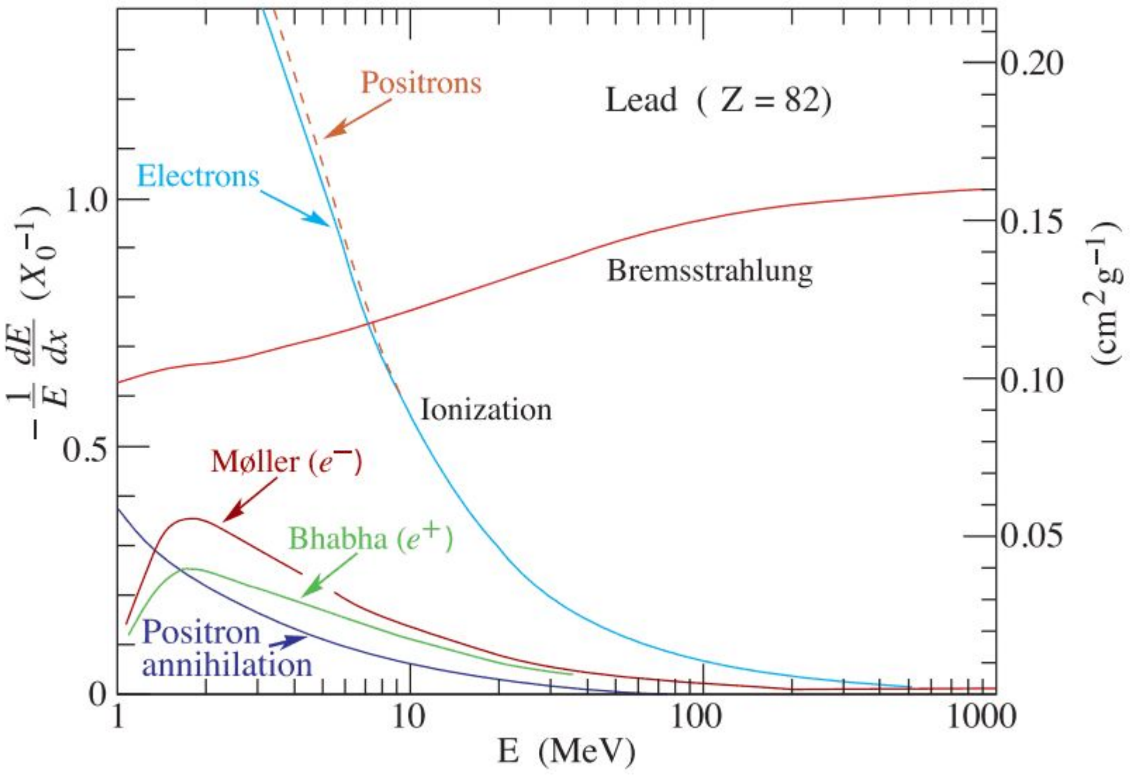
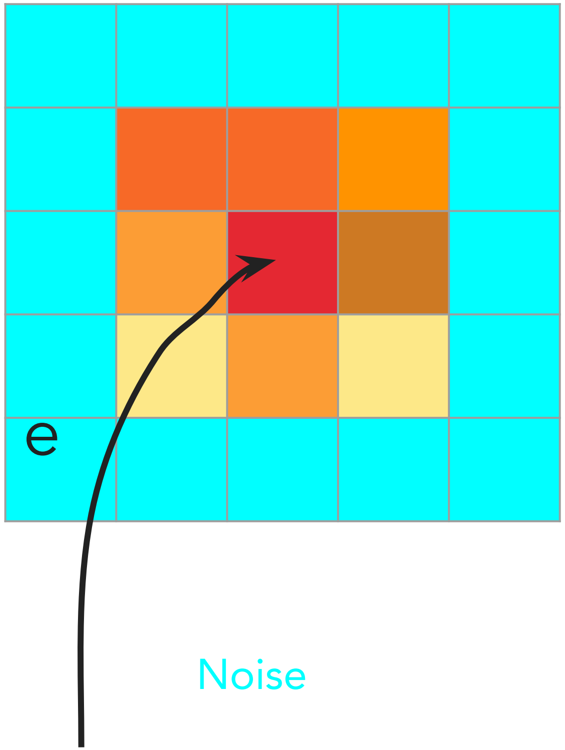


Fig. 5.4: ECAL readout chain.

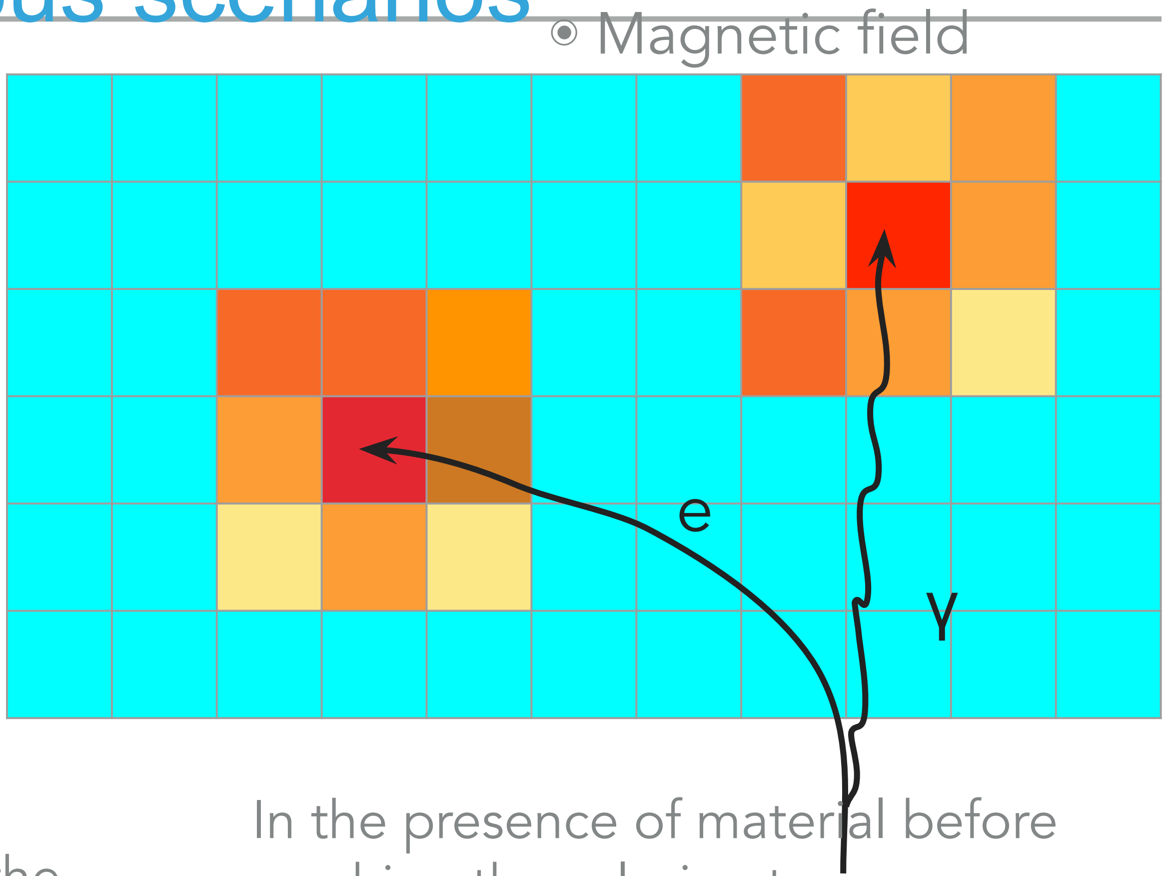



Positrons
Lead (Z = 82)
Electrons
1.0
0.20
0.15
0.10
0.05
Bremsstrahlung
Ionization
0.5
Møller (e^-)
Bhabha (e^+)
Positron annihilation
0
1
10
100
1000
E (MeV)
$-\frac{1}{E}\frac{dE}{dx}$ (X_0^{-1})
($cm^2 g^{-1}$)

Review of various scenarios



No material placed before the calorimeter



In the presence of material before reaching the calorimeter

Photon should be tangent to the electron line → could not do better