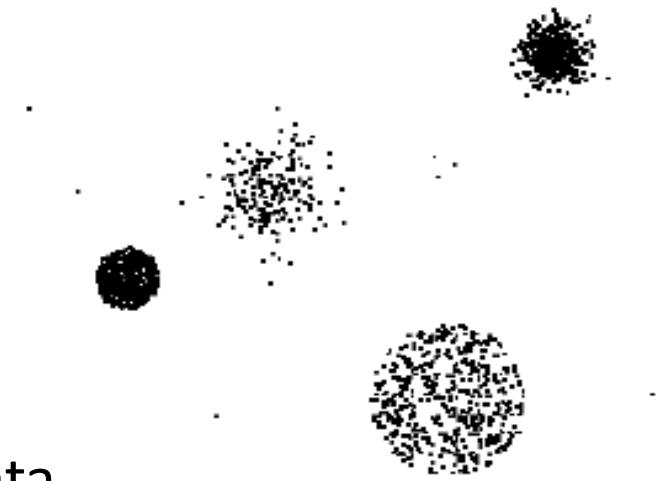
Anomaly Detection

Anomaly/Outlier Detection

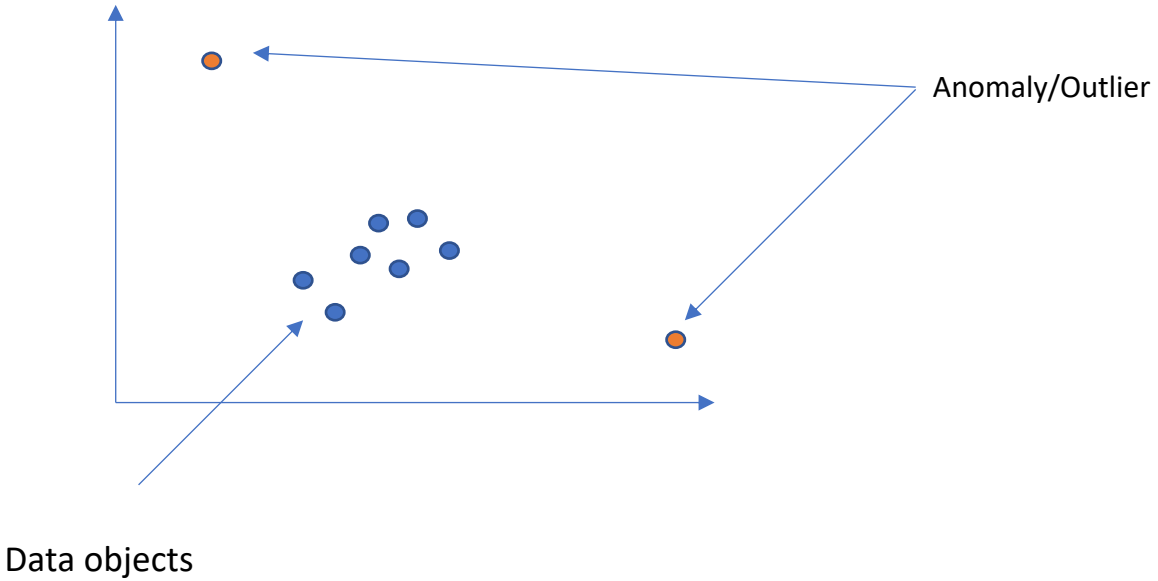
- What are anomalies/outliers?
  - The set of data points that are considerably different than the remainder of the data.
- Natural implication is that anomalies are relatively rare
  - One in a thousand occurs often if you have lots of data
  - Context is important, e.g., freezing temps in July
- Can be important or an error
  - Unusually high blood pressure
  - 200 pound, 2 year old



Anomaly/Outlier Detection

- **Anomaly detection**

  The goal is to find objects that don’t conform to normal patterns of behaviour.



Anomaly Detection

- **Deviation Detection**. This is another name for anomaly detection, anomalous objects have attribute values that deviate significantly from the expected typical attribute values.
- All definitions capture the notion that an anomalous object is unusual in some way or inconsistent with other objects.
- Although unusual objects or events are, by definition, relatively rare, their detection and analysis provides critical insights that are useful in a number of applications.

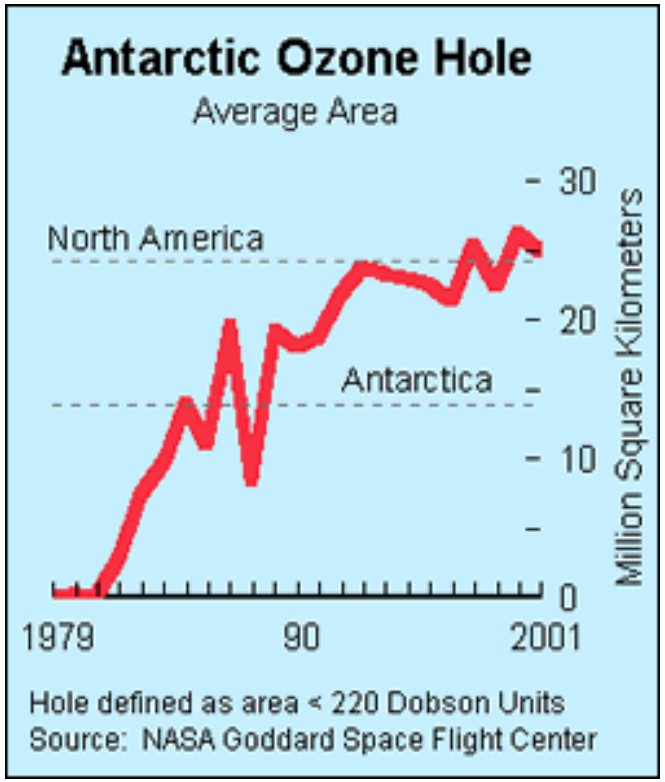
Anomaly Detection

- Much of the recent interest in anomaly detection is driven by applications in which anomalies are the focus.
- However, historically, anomaly detection (and removal) has been viewed as a data pre-processing technique to eliminate erroneous data objects that may be recorded because of human error, a problem with the measuring device, or the presence of noise.
- Such anomalies provide no interesting information but only distort the analysis of normal objects.

Importance of Anomaly Detection

Ozone Depletion History

- In 1985 three researchers (Farman, Gardinar and Shanklin) were puzzled by data gathered by the British Antarctic Survey showing that ozone levels for Antarctica had dropped 10% below normal levels.
- Why did the Nimbus 7 satellite, which had instruments aboard for recording ozone levels, not record similarly low ozone concentrations?
- The ozone concentrations recorded by the satellite were so low they were being treated as outliers by a computer program and discarded!



Causes of Anomalies

- Data from different classes
  - Measuring the weights of oranges, but a few grapefruit are mixed in
- Natural variation
  - Unusually tall people
- Data errors
  - 200 pound 2 year old

Distinction between noise and anomalies

- Noise doesn’t necessarily produce unusual values or objects
- Noise is not interesting
- Noise and anomalies are related but distinct concepts

Applications

- **Fraud Detection.**

  Purchasing behaviour of someone who has a stolen credit card is often different to the behaviour of the original owner of the credit card.

  Credit card companies attempt to detect a theft by **looking at buying patterns** that characterise theft or by noticing change from typical behaviour.

Applications

- **Medicine and Public Health**

  For a particular patient unusual symptoms such as test results or anomalous MRI scans may indicate potential health problems. The characterization of something as anomalous or not incurs costs. This may involve extra tests for the patient if the patient is healthy and harm to the patient if the condition is left undiagnosed.

  Detecting of emerging disease outbreak such as H1N1 influenza, SARS and Covid-19 which results in unusual test results in a series of patients. This also important to monitoring the spread of disease and taking preventative action.

Applications

- **Aviation Safety.**

  Since aircrafts are highly complex and dynamic systems, they are **prone to accidents** due to mechanical, environmental or human factors.

  - To **monitor the occurrence of such anomalies**, most commercial airplanes are **equipped with a large number of sensors** to measure different flight parameters, such as information from the control system, the avionics and propulsion systems, and pilot actions.
  - **Identifying abnormal events** in these sensor recordings (e.g., an anomalous sequence of pilot actions or an abnormally functioning aircraft component) can help prevent aircraft accidents and promote aviation safety.

Applications

- **Independent Living.**

  Monitoring housing environment through sensors of elderly or vulnerable adults living independently. For example several sensors embedded at key positions inside the house to monitor the individual's behaviour.

  Anomalous behaviour (such as the individual hasn't moved for several hours) will be flagged as a case for concern.

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique.
- Some of the common characteristics of the input data include:
  - the **number** and **types of attributes**, and
  - the **representation** used for describing every data instance.

Univariate and Multivariate Data

Data can be univariate and multivariate.

- **Univariate** – The data contains a single attribute. A data object is anomalous if its value is different to the other normal attribute values.
- **Multivariate –** The data is represented by many attributes, the data may be anomalous for some but normal for others . A data object with many attributes may not have anomalous values but together they may be anomalous.

  Identifying an anomaly in a multivariate setting is challenging particularly when the dimensionality of the data is high.

Availability of labels

- In most practical applications, we do **not have a training set with accurate** and **representative labels** of the **normal** and **anomalous** classes.
- Obtaining **labels of the anomalous** class is especially challenging because of their **rarity**.
- Hence, most anomaly detection problems are unsupervised in nature; i.e. the input data does not have any labels.
- However, anomalies typically have some properties that techniques can take advantage of to make finding anomalies practical:
  - relatively small in number;
  - sparsely distributed.

General Issues: Label vs Score

- Different approaches for anomaly detection produce their outputs in different formats.
- The most basic type of output is a binary anomaly label: an object is either identified as an anomaly or as a normal instance.
- However, labels do not provide any information about the degree to which an instance is anomalous.
- Frequently, some of the detected anomalies are more extreme than others, while some instances labeled as normal may be on the verge of being identified as anomalies.

Label vs Score

- Hence, many anomaly detection methods produce an **anomaly score** that indicates **how strongly an instance is likely** to be an **anomaly.**
- An **anomaly score** can easily be **sorted and converted** into **ranks**, so that an analyst can be provided with only the **top-most scoring anomalies.**
- Alternatively, a **cutoff threshold** can be applied to an **anomaly score** to obtain **binary anomaly labels.**
- The task of **choosing the right threshold** is often left to the **discretion of the analyst.**
- However, sometimes the **scores** have an **associated meaning**,
  - e.g. **statistical significance**, which makes the analysis of anomalies easier and more interpretable.

Anomaly Detection Techniques

- **Statistical Approaches**
- **Machine Learning Approaches**
  - Proximity-based
    - Anomalies are points far away from other points
  - Clustering-based
    - Points far away from cluster centers are outliers
    - Small clusters are outliers
  - Ensemble-based approaches
  - Reconstruction Based

Statistical Approaches

Statistical Approaches

- **Probabilistic definition of an outlier:** an outlier is an object that has a low probability with respect to a probability distribution model of the data.
- Statistical approaches make use of probability distributions (e.g., the Gaussian distribution) to model the normal class.
- A key feature of such distributions is that they associate a probability value to every data instance, indicating how likely it is for the instance to be generated from the distribution.
- Anomalies are then identified as instances that are unlikely to be generated from the probability distribution of the normal class.

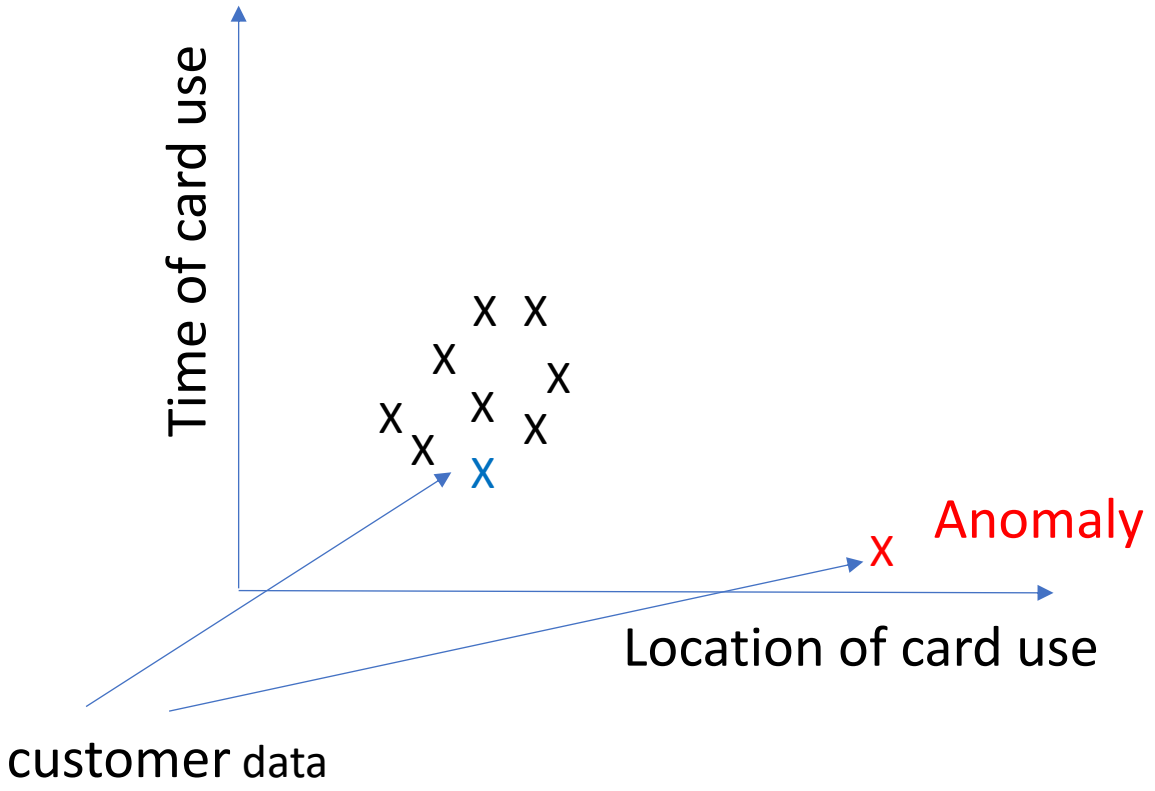
Model-based Anomaly Detection Example

Fraud Detection

Given a data set $\{x^1, x^2, x^3, \ldots x^n\}$

The customer "features" based on credit card use are:

- x_1 = time of card use
- x_2 = location of card use
- x_3 = amount used with card



Model-based Anomaly Detection

- We can identify user transactions that are different.

  We need to find a model $P(X)$ and a probability threshold ε.

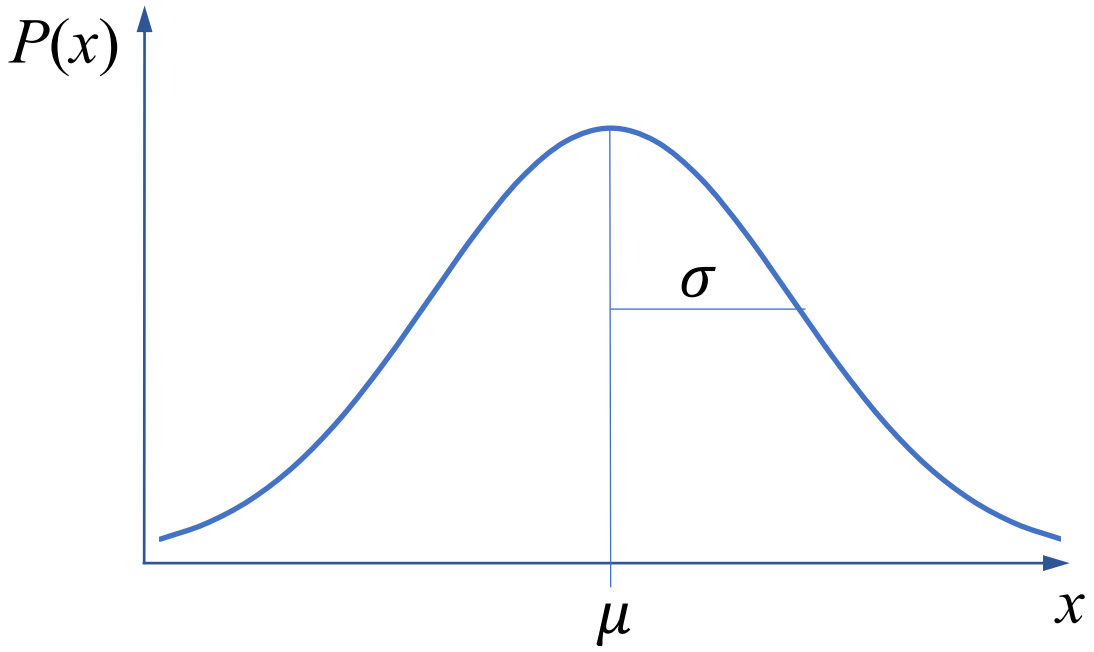
  If $P(X) \geq \varepsilon$, then this is similar to the pattern of normal customer behaviour.

  If $P(X) < \varepsilon$ then this is anomalous.

Gaussian (Normal) Distribution

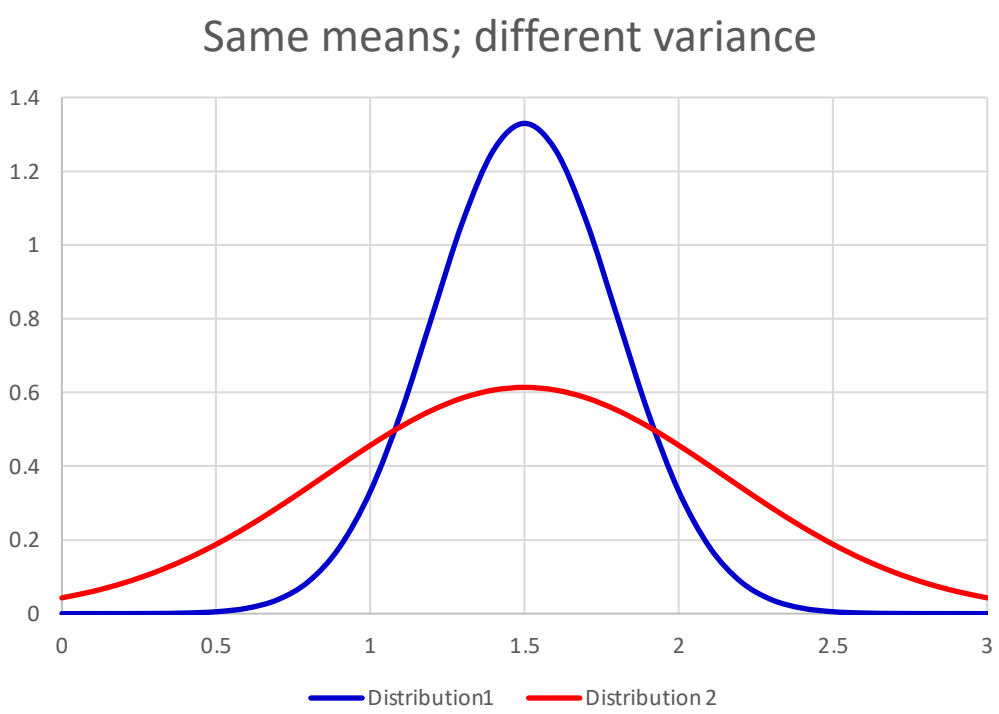
Assume $x \in \mathbb{R}$. If x has a Gaussian (Normal) distribution with a mean μ and variance σ^2, we write $x \sim \mathcal{N}(\mu, \sigma^2)$.

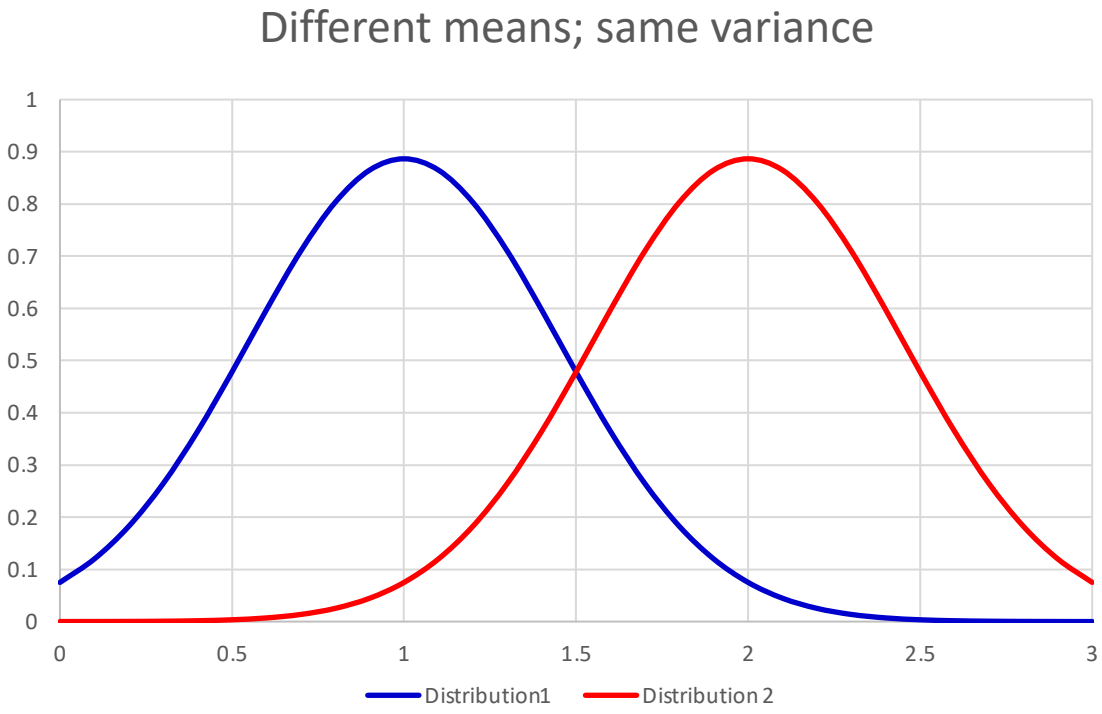
The Gaussian distribution looks like a bell shaped curve, centred on the mean.



$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(x-\mu)^2}{2\sigma^2})}$$

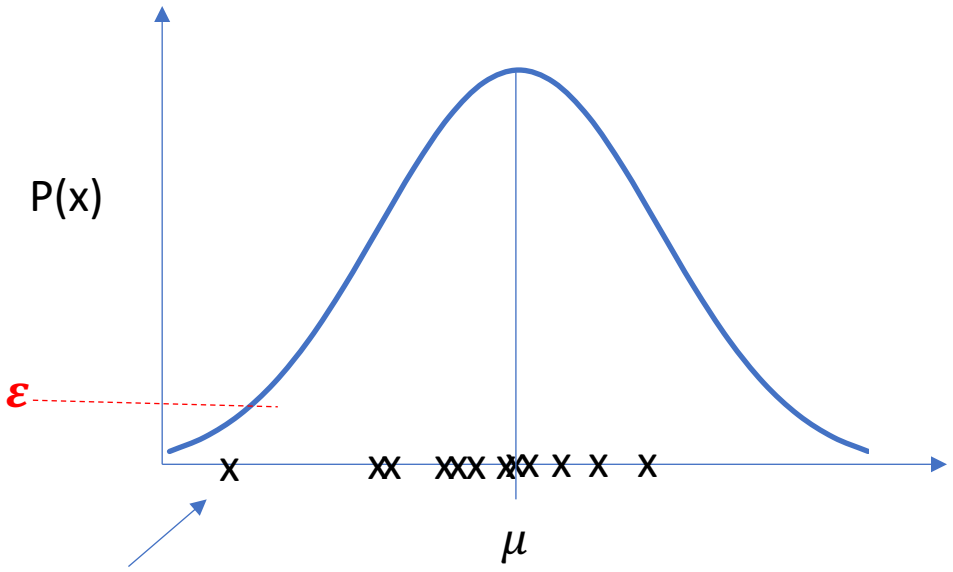
Normal distribution: varying mean and standard deviation





Univariate model

Say we have an example dataset $\{x^1, x^2, x^3, \ldots x^n\}$ and if we are to plot these x values.



$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^i$$

$$\sigma^2 = \frac{1}{m}\sum_{i=1}^{m} (x^i - \mu)^2$$

Calculating the Normal Distribution using Python

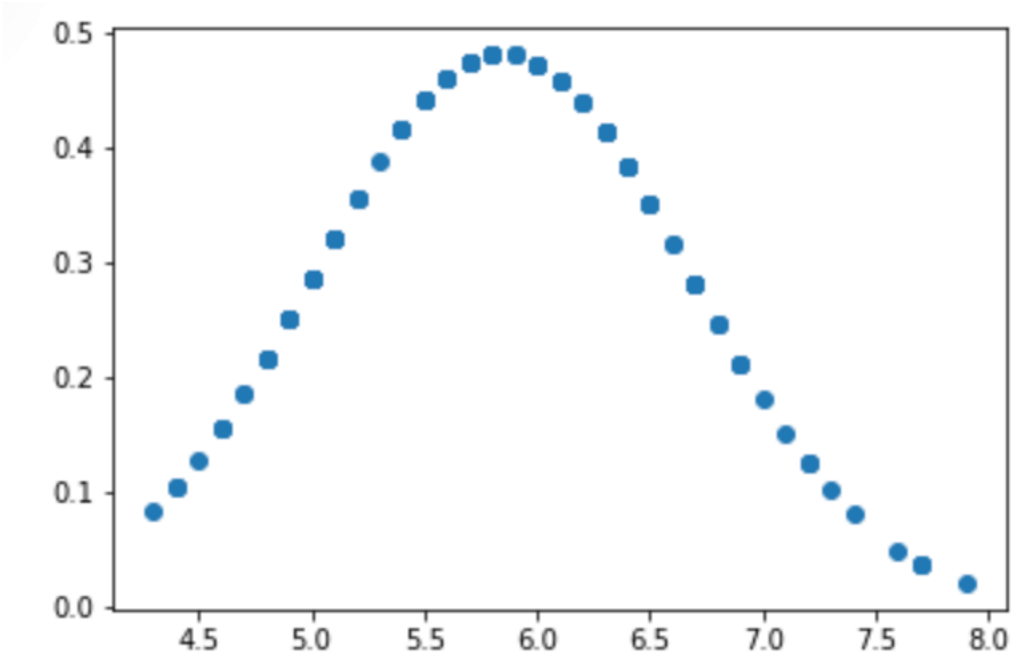
```
from scipy  import stats

stats.norm.pdf(x, loc=mean, scale=sd)  #x is the data, mean and sd are
                                       #mean and standard deviation
                                       #of the data
```

Example - Iris dataset

```
from scipy  import stats
import pandas as pd
import matplotlib.pyplot as plt

df=pd.read_csv('iris.csv’)
m_sl=df['sepal_length'].mean()
sd_sl=df['sepal_length'].std()

stats.norm.pdf(df['sepal_length'],m_sl,sd_sl)
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...	...	...	...	...	...

```
a=df['sepal_length'].values
plt.scatter(a, stats.norm.pdf(df['sepal_length'],m_sl,sd_sl))
plt.show()
```

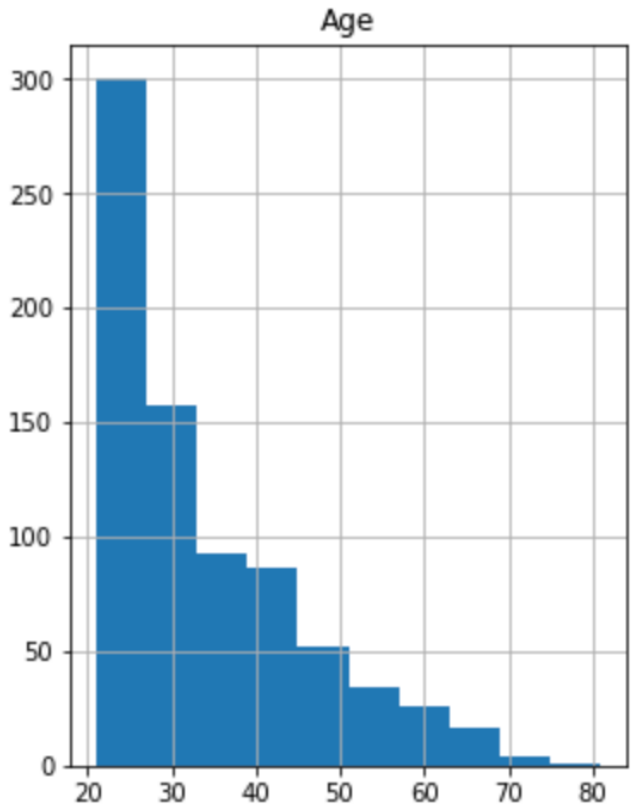


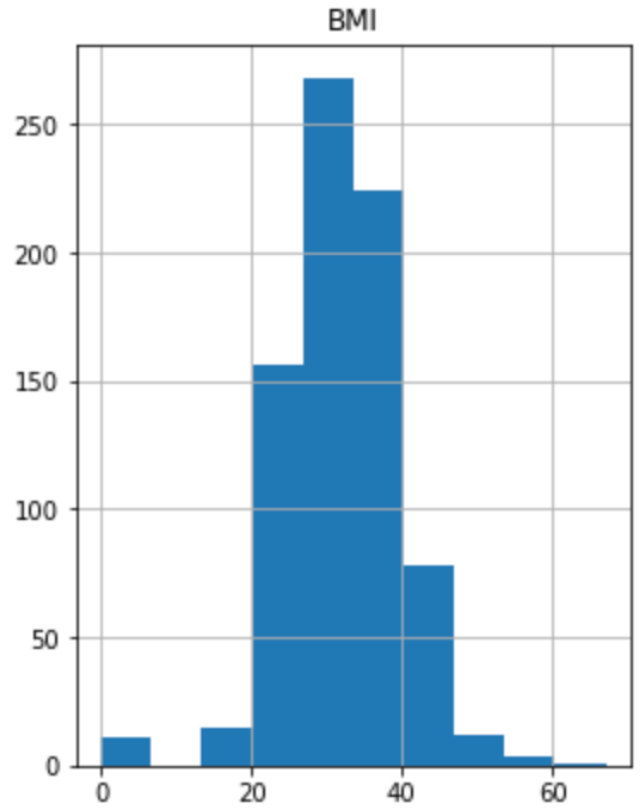
- Continuous data typically follow a Gaussian (Normal) distribution. A way to check this is to visualize the data using histograms. In python this is done using:

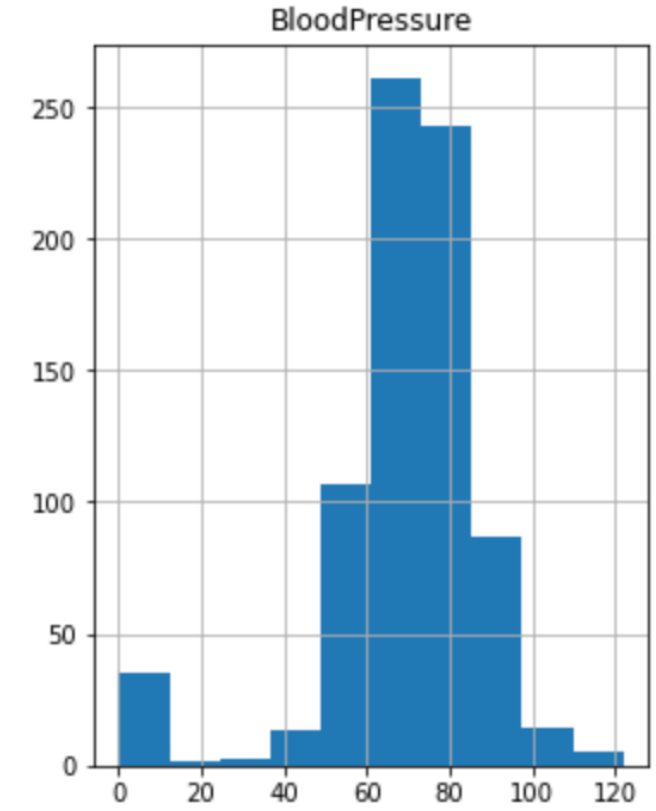
```
from matplotlib import pyplot as plt

    df.hist(figsize=(15,20))
    plt.show()
```

Histograms

We get the following output

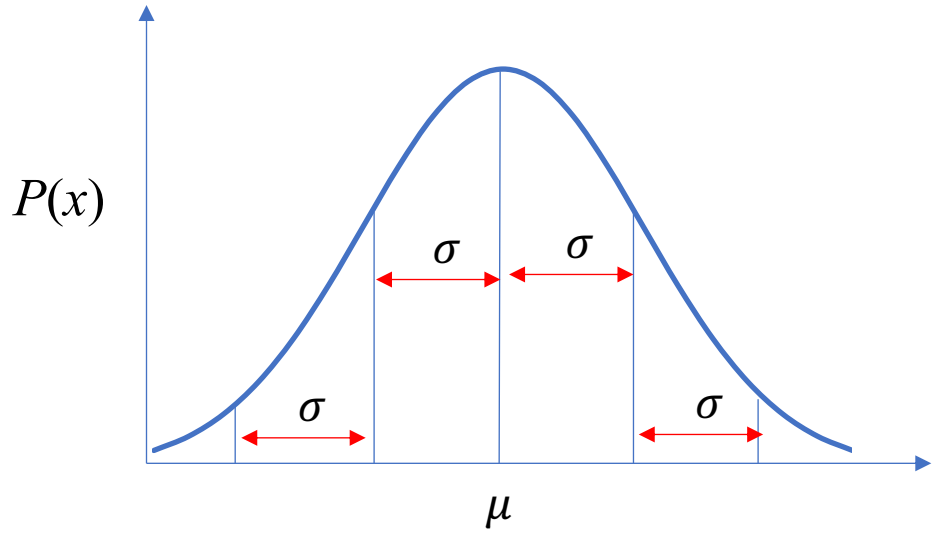






Identifying outliers using a normal distribution

- For a Gaussian distribution, 95% of the data lies within two standard deviations of the mean.
- So, 2.5% of the data is greater than 2 standard deviations above the mean and 2.5% of the data is less than 2 standard deviations below the mean.



Statisticians frequently regard data as anomalous if it is 2 standard deviations from the mean.

Getting the threshold

```
df['outlier']=0
df['outlier'] += ((df['sepal_length’] > df['sepal_length'].mean()+2*df['sepal_length'].std()) | (df['sepal_length'] < df['sepal_length'].mean() - 2*df['sepal_length'].std())).astype(int)
```

```
df[df['outlier']>0]
```

	sepal_length	sepal_width	petal_length	petal_width	species	outlier
105	7.6	3.0	6.6	2.1	virginica	1
117	7.7	3.8	6.7	2.2	virginica	1
118	7.7	2.6	6.9	2.3	virginica	1
122	7.7	2.8	6.7	2.0	virginica	1
131	7.9	3.8	6.4	2.0	virginica	1
135	7.7	3.0	6.1	2.3	virginica	1

Multivariate Model (assuming independence)

- The data set $\{x^1, x^2, x^3, \ldots, x^n\}$ – each x is a feature vector. For example, with fraud detection x^1 would be transactions of customer 1 etc. So each example is $x^i \in \mathbb{R}^n$.
- We need to model $P(x)$ from the datasets where x is a vector.
- An assumption is made that all features are independent:

$$P(x) = P(x_1; \mu_1, \sigma_1^2).\, P(x_2; \mu_2, \sigma_2^2).P(x_3; \mu_3, \sigma_3^2)\ldots\, P(x_n; \mu_n, \sigma_n^2).$$

$$P(x) = \prod_{j=1}^{n} P\left(x_j; \mu_j, \sigma_j^2\right)$$

Anomaly Detection Algorithm

- Choose features x_i that may be indicative of anomalous behaviour. Assume the features are independent.
- Fit parameters $\mu_1, \dots, \mu_n, \sigma_1^2, \dots, \sigma_n^2$:

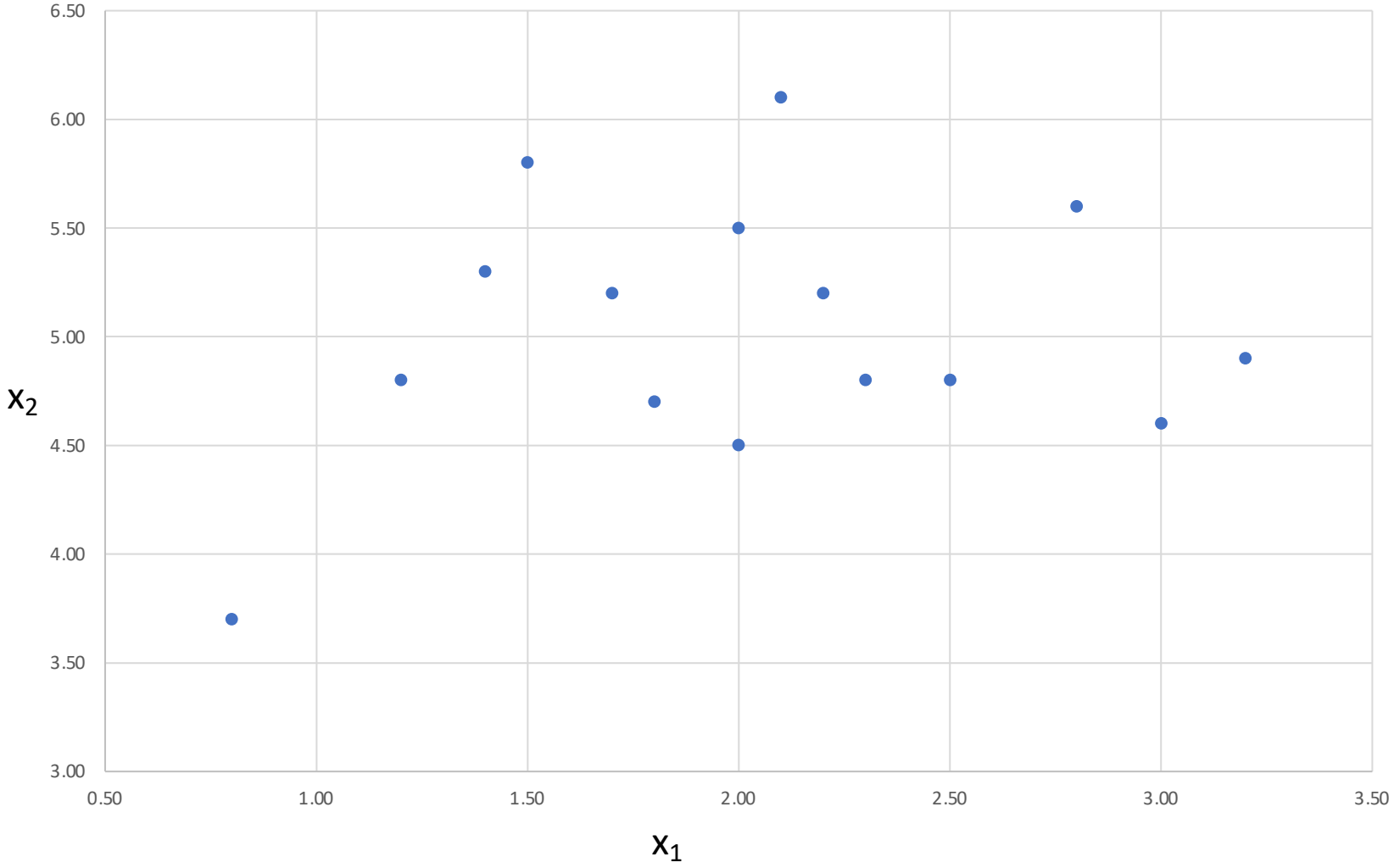
$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_j^i, \quad \sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}\left(x_j^i - \mu_j\right)^2.$$

- Given a data object x, compute $P(x)$. If $P(x) < \varepsilon$ then it is an anomaly.

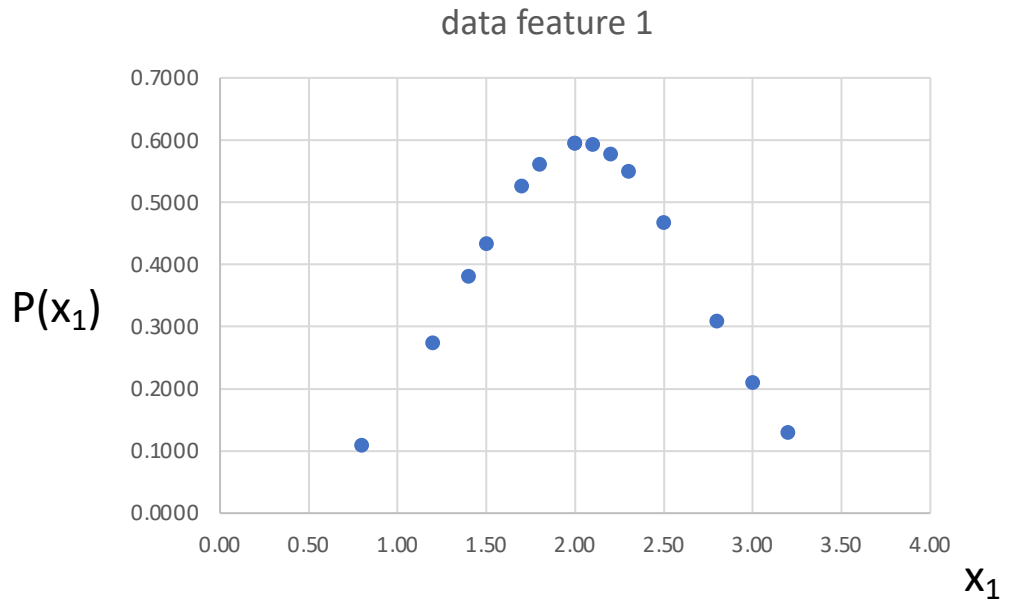
$$P(x) = \prod_{j=1}^{n} P\left(x_j; \mu_j, \sigma_j^2\right) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\,\sigma_j} exp^{\left(-\frac{\left(x_j-\mu_j\right)^2}{2\sigma_j^2}\right)}$$

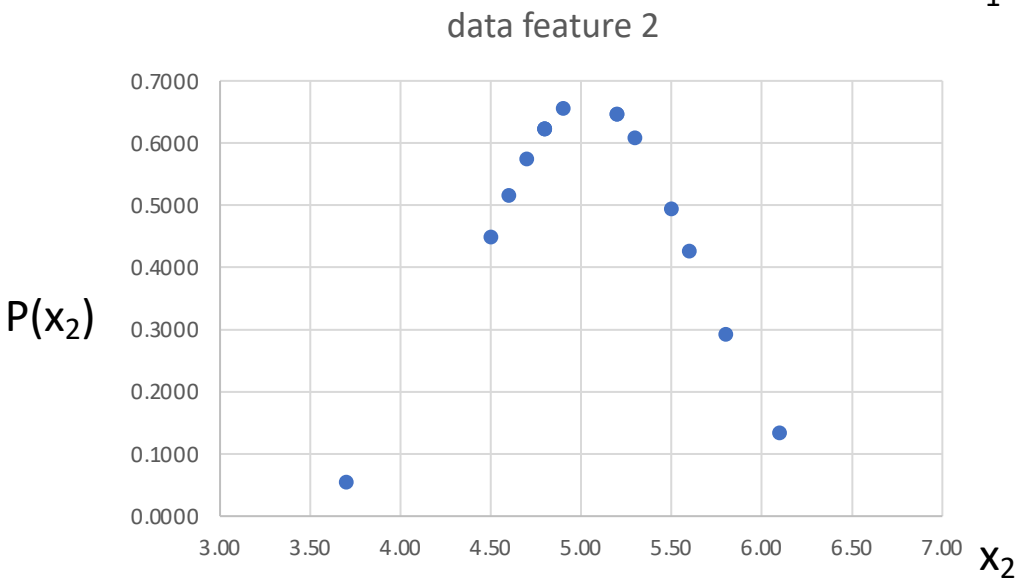
Example of Anomaly Detection with Gaussian Distribution (Bivariate)

		Features	
		x_1	x_2
items	$x^{(1)}$	2.00	5.50
	$x^{(2)}$	2.50	4.80
	$x^{(3)}$	3.00	4.60
	$x^{(4)}$	2.10	6.10
	$x^{(5)}$	1.80	4.70
	$x^{(6)}$	1.50	5.80
	$x^{(7)}$	0.80	3.70
	$x^{(8)}$	2.30	4.80
	$x^{(9)}$	2.80	5.60
	$x^{(10)}$	3.20	4.90
	$x^{(11)}$	2.20	5.20
	$x^{(12)}$	2.00	4.50
	$x^{(13)}$	1.40	5.30
	$x^{(14)}$	1.70	5.20
	$x^{(15)}$	1.20	4.80
	Mean m	2.0333	5.0333
	Variance s^2	0.4481	0.3524



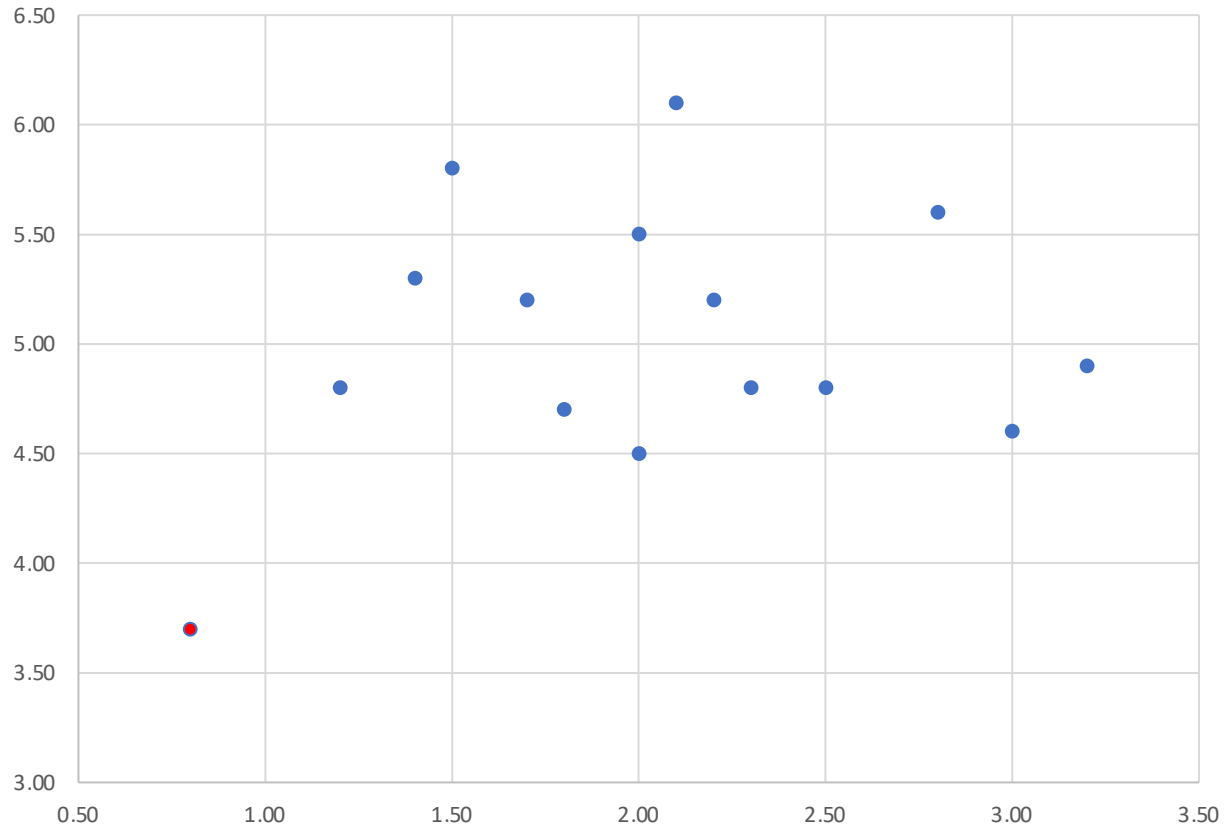
		Features			
		x_1	x_2	$p(x_1)$	$p(x_2)$
Items	$x^{(1)}$	2.00	5.50	0.5952	0.4934
	$x^{(2)}$	2.50	4.80	0.4674	0.6221
	$x^{(3)}$	3.00	4.60	0.2101	0.5149
	$x^{(4)}$	2.10	6.10	0.5930	0.1337
	$x^{(5)}$	1.80	4.70	0.5608	0.5740
	$x^{(6)}$	1.50	5.80	0.4339	0.2919
	$x^{(7)}$	0.80	3.70	0.1092	0.0539
	$x^{(8)}$	2.30	4.80	0.5505	0.6221
	$x^{(9)}$	2.80	5.60	0.3093	0.4261
	$x^{(10)}$	3.20	4.90	0.1305	0.6553
	$x^{(11)}$	2.20	5.20	0.5778	0.6461
	$x^{(12)}$	2.00	4.50	0.5952	0.4489
	$x^{(13)}$	1.40	5.30	0.3809	0.6076
	$x^{(14)}$	1.70	5.20	0.5265	0.6461
	$x^{(15)}$	1.20	4.80	0.2746	0.6221
	Mean m	2.0333	5.0333		
	Variance s^2	0.4481	0.3524		
	Sqrt(2p)s	1.6779	1.4880		
	s	0.6694	0.5936		





$$P(x) = P(x_1).P(x_2) \qquad \varepsilon = 0.01$$

		Features					
		x_1	x_2	$p(x_1)$	$p(x_2)$	p(item)	ε = 0.01
Items	$x^{(1)}$	2.00	5.50	0.5952	0.4934	0.2937	ok
	$x^{(2)}$	2.50	4.80	0.4674	0.6221	0.2908	ok
	$x^{(3)}$	3.00	4.60	0.2101	0.5149	0.1082	ok
	$x^{(4)}$	2.10	6.10	0.5930	0.1337	0.0793	ok
	$x^{(5)}$	1.80	4.70	0.5608	0.5740	0.3219	ok
	$x^{(6)}$	1.50	5.80	0.4339	0.2919	0.1266	ok
	$x^{(7)}$	0.80	3.70	0.1092	0.0539	0.0059	anomaly
	$x^{(8)}$	2.30	4.80	0.5505	0.6221	0.3425	ok
	$x^{(9)}$	2.80	5.60	0.3093	0.4261	0.1318	ok
	$x^{(10)}$	3.20	4.90	0.1305	0.6553	0.0855	ok
	$x^{(11)}$	2.20	5.20	0.5778	0.6461	0.3733	ok
	$x^{(12)}$	2.00	4.50	0.5952	0.4489	0.2672	ok
	$x^{(13)}$	1.40	5.30	0.3809	0.6076	0.2314	ok
	$x^{(14)}$	1.70	5.20	0.5265	0.6461	0.3401	ok
	$x^{(15)}$	1.20	4.80	0.2746	0.6221	0.1708	ok
	Mean m	2.0333	5.0333				
	Variance s^2	0.4481	0.3524				
	Sqrt(2p)s	1.6779	1.4880				
	s	0.6694	0.5936				



Multivariate Gaussian (Normal) Distribution

- **Correlated data – assume that features are not independent**.

  For data set comprised of more than one continuous attribute, we can use a multivariate Gaussian distribution to model the normal class.

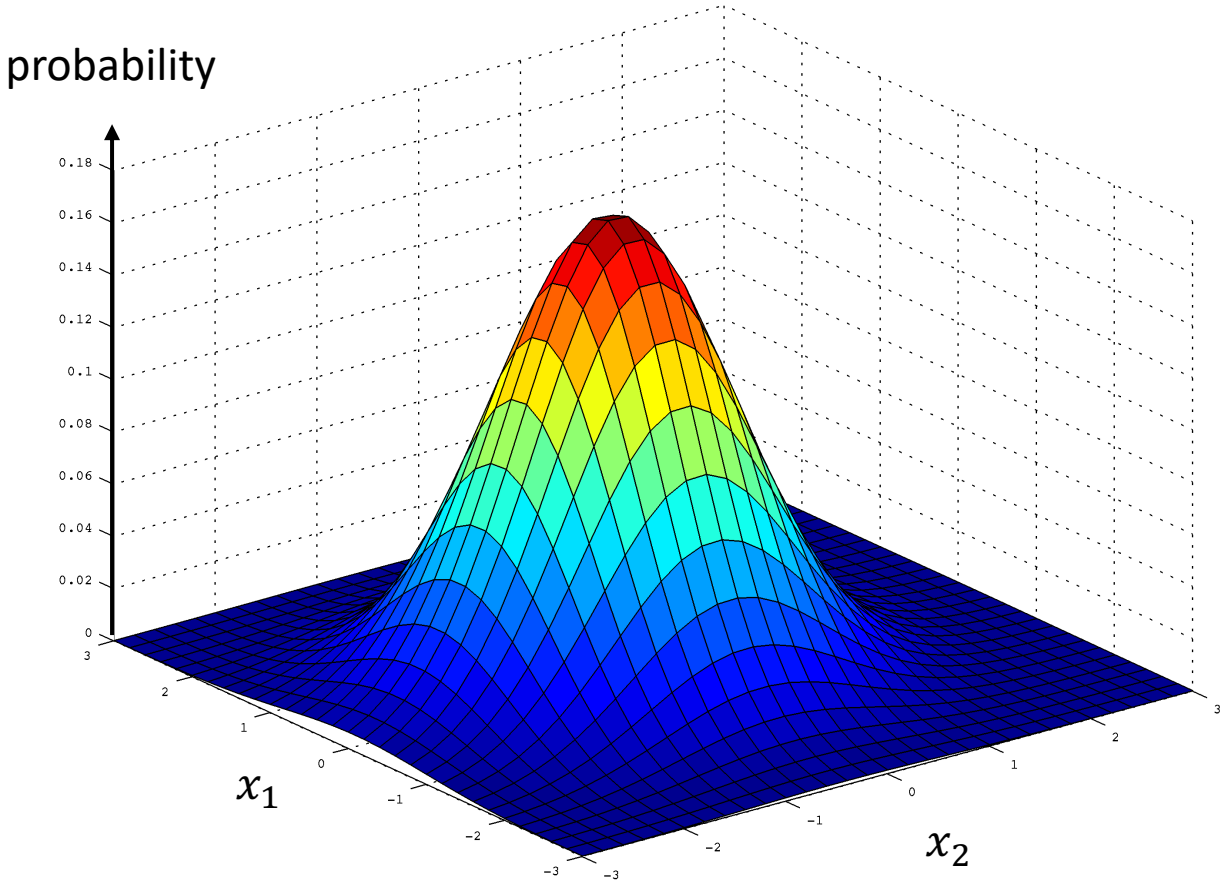
  A multivariate Gaussian distribution $N(\mu, \Sigma)$ involves the parameters μ and the covariance matrix Σ, which need to be estimated from the data.

  The probability density function of a point x as $N(\mu, \Sigma)$ is given by

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

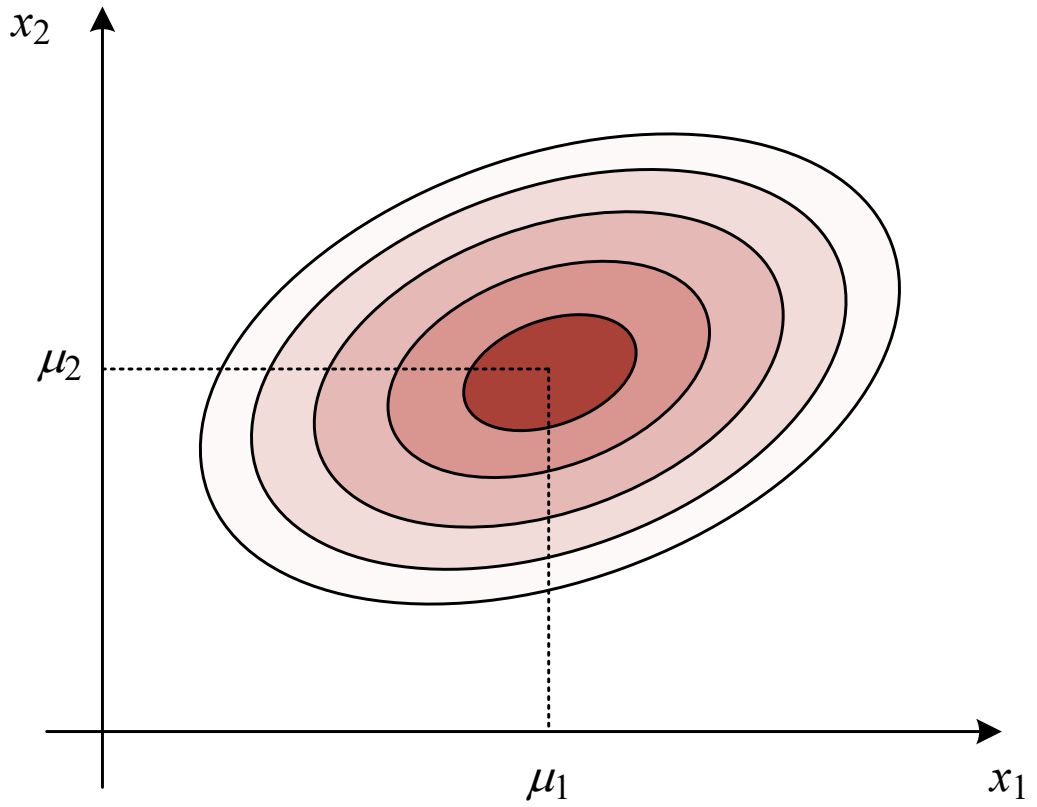
where, $|\Sigma|$ is the determinant of the covariance matrix Σ and m is the number of dimensions of x.

Bivariate normal distribution

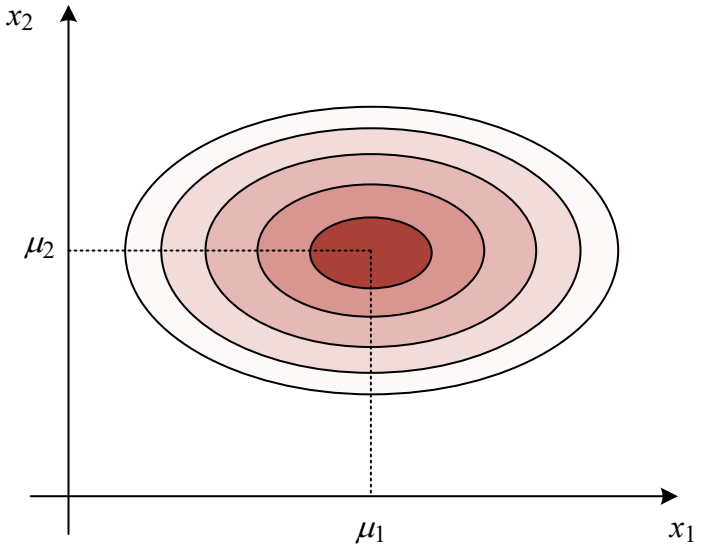


Bivariate normal distribution level curves

Curves of constant probability – ‘slices’ through the probability density function.

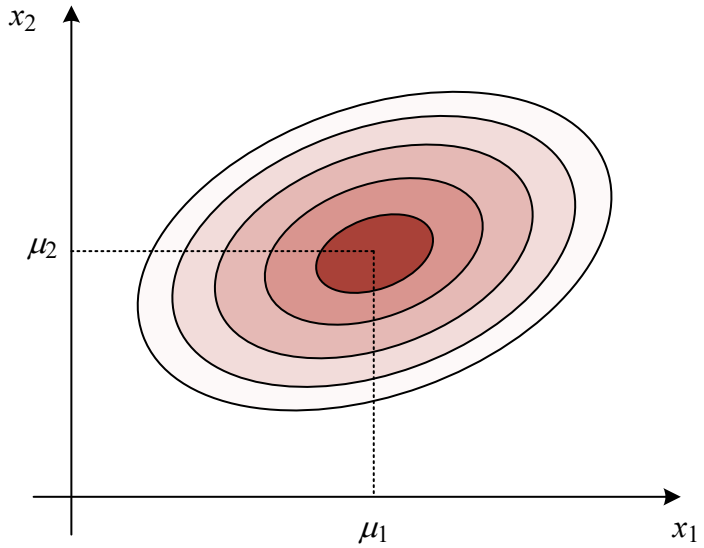


Bivariate normal distribution level curves

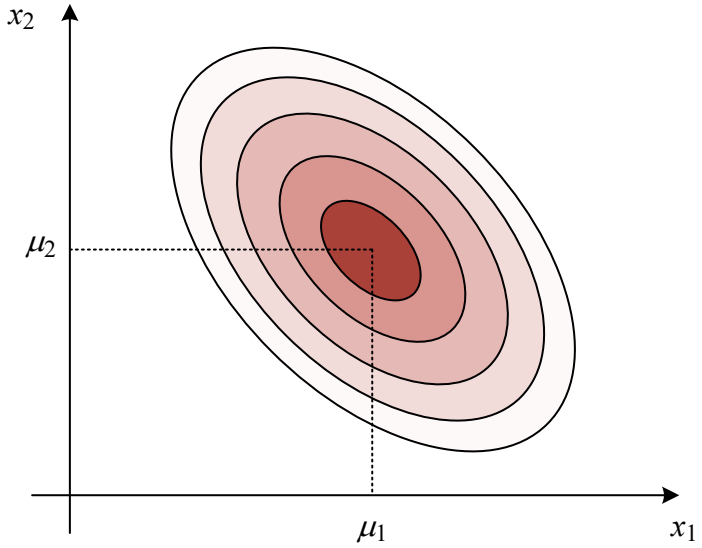


x_1 and x_2 independent

$\sigma_2^2 < \sigma_1^2$



x_1 and x_2 positively correlated



x_1 and x_2 negatively correlated

Covariance Matrix

The covariance matrix for two variables is

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{1,2} \\ \sigma_{1,2} & \sigma_2^2 \end{pmatrix}$$

where, σ_1^2 and σ_2^2 are the variances of of x_1 and x_2 and $\sigma_{1,2}$ is the covariance of x_1 and x_2 defined by

$$\sigma_{1,2} = \frac{1}{m} \sum_{i=1}^{m} (x_1 - \mu_1)\,(x_2 - \mu_2)$$

Covariance Matrix

Calculating the covariance matrix

	X	X^2	Y	Y^2	X.Y
Object 1	1	1	2	4	2
Object 2	5	25	3	9	15
Object 3	3	9	4	16	12
Mean	3	35/3	3	29/3	29/3

$$\sigma_1^2 = E(X^2) - E(X).E(X)$$

$$\sigma_2^2 = E(Y^2) - E(Y).E(Y)$$

$$\sigma_{1,2} = cov(XY) = E(XY) - E(X).E(Y)$$

where, $E(X) = \frac{1}{m}\sum_{i=1}^{m} x_i$

$$E(Y) = \frac{1}{m}\sum_{i=1}^{m} y_i$$

$$E(XY) = \frac{1}{m}\sum_{i=1}^{m} x_i\, y_i$$

$$E(X^2) = \frac{1}{m}\sum_{i=1}^{m} x_i^2$$

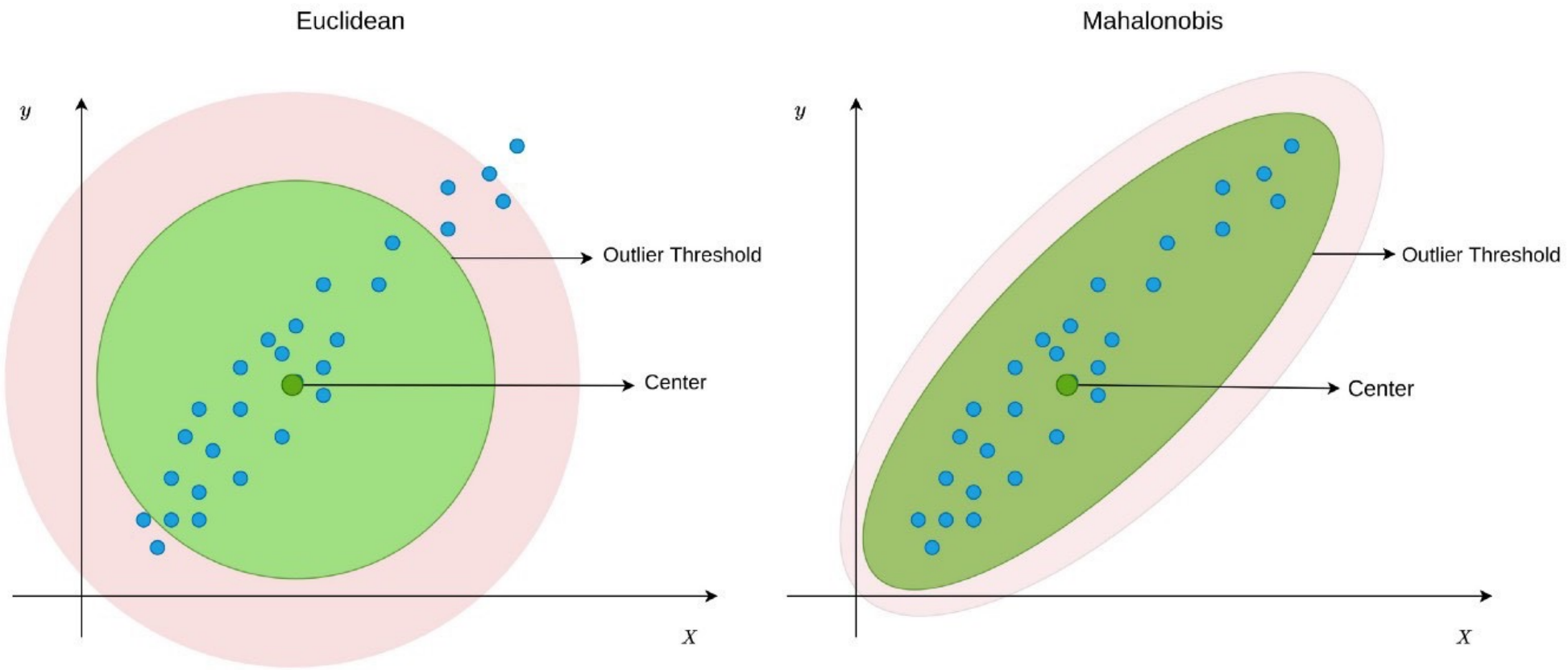
$$E(Y^2) = \frac{1}{m}\sum_{i=1}^{m} y_i^2$$

$$\sigma_1^2 = \frac{35}{3} - 3^2 = \frac{8}{3} \qquad \sigma_2^2 = \frac{29}{3} - 3^2 = \frac{2}{3} \qquad \sigma_{1,2} = \frac{29}{3} - 3\times 3 = \frac{2}{3}$$

Using Multivariate Gaussian Distribution

- In the case of a multivariate Gaussian distribution, the distance of a point from the centre cannot be directly used as a viable anomaly score.
- This is because a multivariate normal distribution is not symmetrical with respect to its centre if there are correlations between the attributes.
- The probability density varies asymmetrically as we move outward from the centre in different directions.
- We need a distance measure that takes the shape of the data into consideration.

Euclidean vs Mahalonobis Distance



Mahalanobis Distance

The distance between two points x and y is

$$Mahalanobis(x, y) = (x - y)^T S^{-1}(x - y)$$

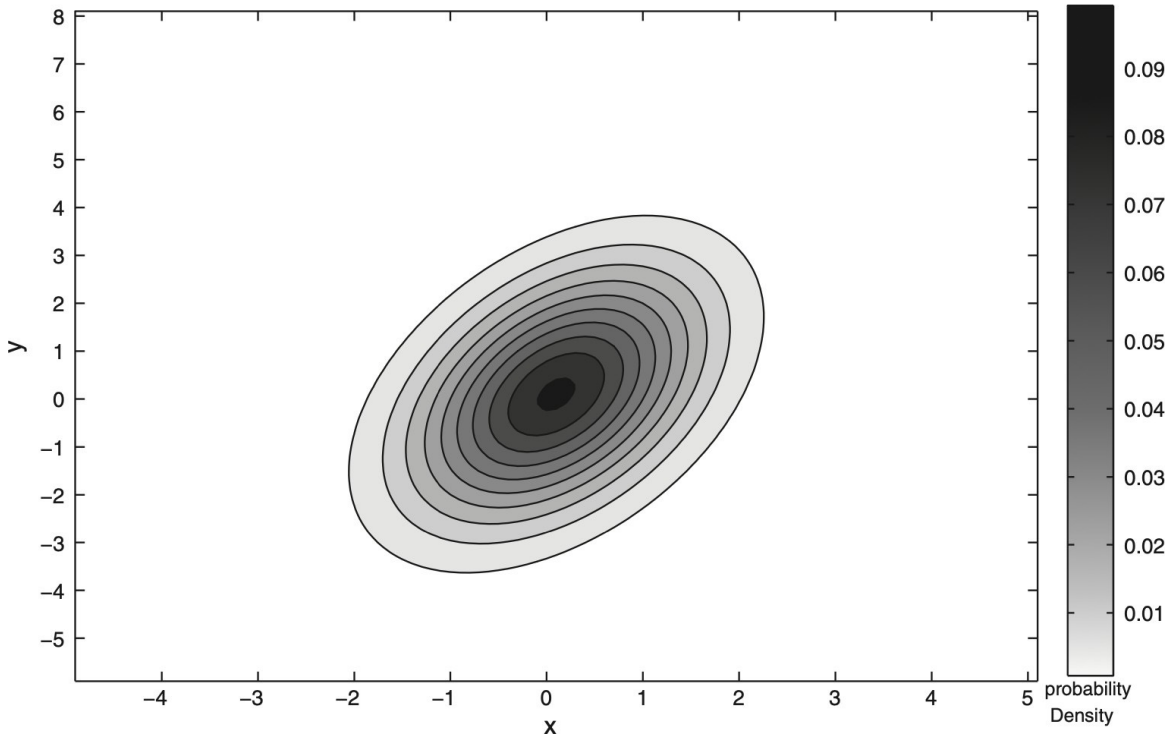
where S is the estimated covariance matrix of the data.

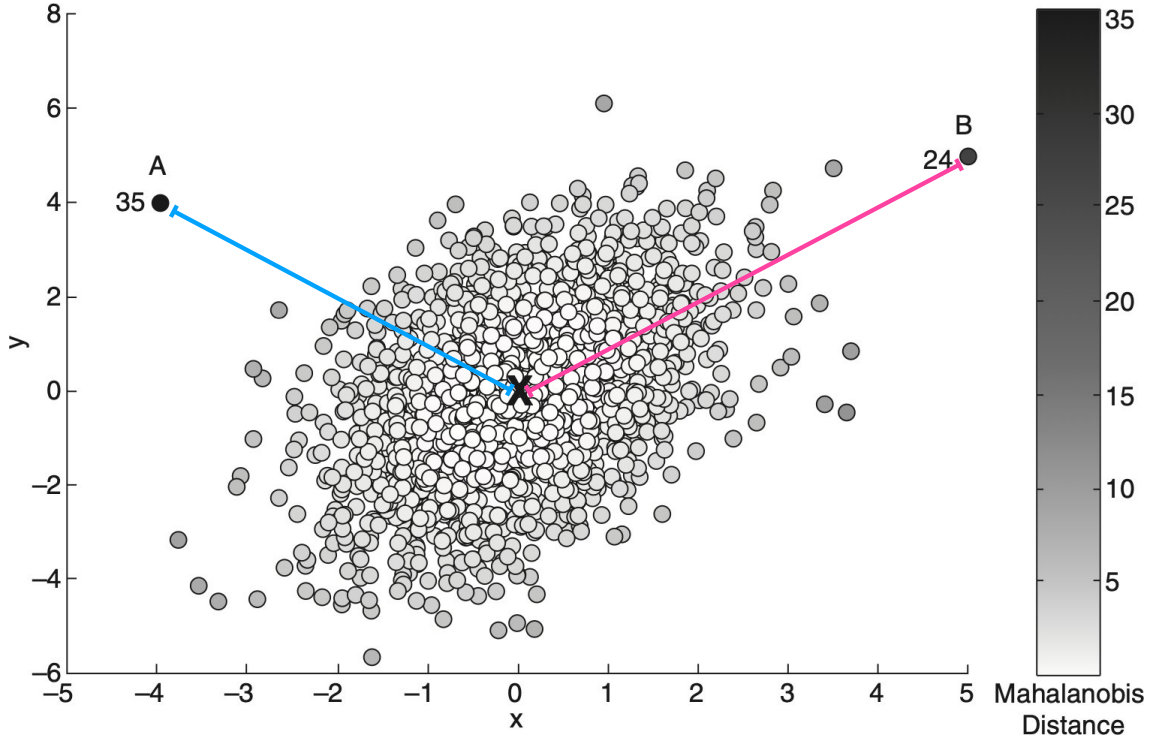
The Mahalanobis distance between x and y is directly related to probability density.

$$f(x) = \frac{1}{\left(\sqrt{2\pi}\right)^m |\Sigma|^{1/2}} e^{-\frac{(x-\mu)^T \Sigma^{-1}(x-\mu)}{2}}$$

Mahalanobis Distance

Even though **A** is closer to the centre as measured by Euclidean distance, it is further away than **B** in terms of the Mahalanobis distance because the Mahalanobis distance considers the shape of the distribution.





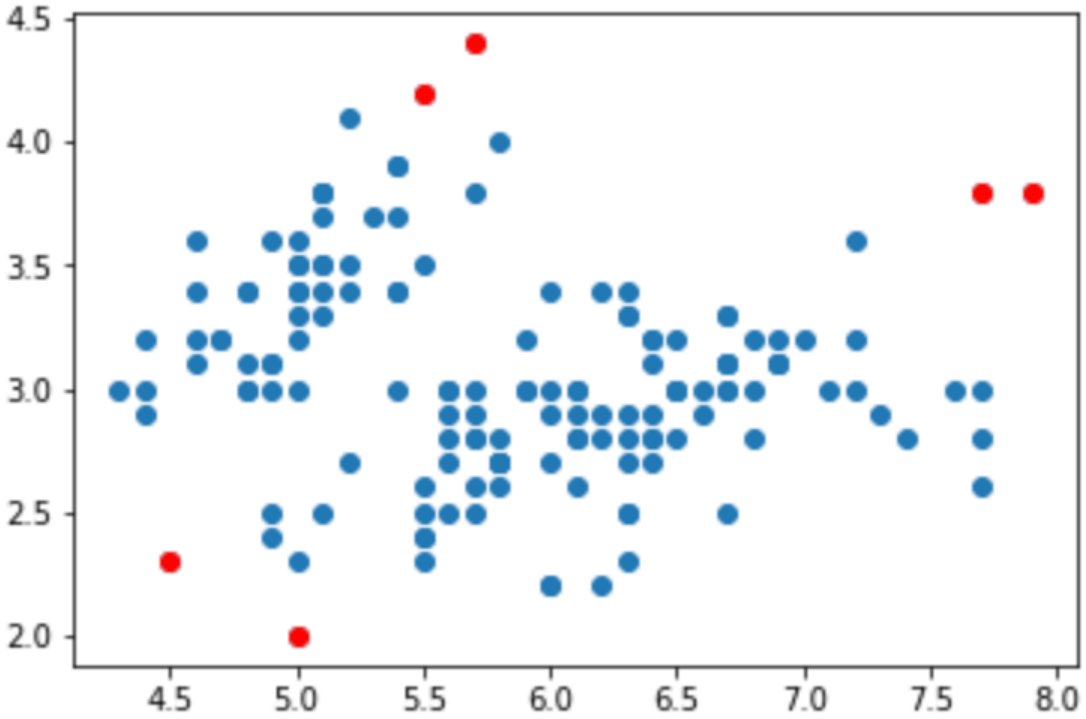
Multivariate gaussain in Python

```
from scipy.stats import multivariate_normal
import numpy as np
df2=pd.read_csv('iris.csv’)
covariance_matrix = np.cov(df2['sepal_length'],df2['sepal_width’])
mean_values=[np.mean(df2['sepal_length']),np.mean(df2['sepal_width
’])]
data=df2[['sepal_length','sepal_width']]
```

```
model=multivariate_normal.pdf(data,cov=covariance_matrix,mean=mean_values)
df2['pdf']=model
b=df2[df2['pdf']<0.02]
```

	sepal_length	sepal_width	petal_length	petal_width	species	pdf
0	5.1	3.5	1.4	0.2	setosa	0.195019
1	4.9	3.0	1.4	0.2	setosa	0.223922
2	4.7	3.2	1.3	0.2	setosa	0.168834
3	4.6	3.1	1.5	0.2	setosa	0.143387
4	5.0	3.6	1.4	0.2	setosa	0.139114
...	...	...	...	...	...	...

```
plt.scatter(df2['sepal_length'],df2['sepal_width'])
plt.scatter(b['sepal_length'],b['sepal_width'], color='r')
```



Strengths and Weaknesses

- Statistical approaches to outlier detection have a firm theoretical foundation and build on standard statistical techniques.
- When there is sufficient knowledge of the data and the type of test that should be applied, these approaches are statistically justifiable and can be very effective.
- They can also provide confidence intervals associated with the anomaly scores, which can be very helpful in making decisions about test instances.
  - e.g., determining thresholds on the anomaly score.

Strengths and Weaknesses

- However, if the wrong model is chosen, then a normal instance can be erroneously identified as an outlier.
- Furthermore, while there are a wide variety of statistical outlier tests for single attributes, far fewer options are available for multivariate data, and these tests can perform poorly for high-dimensional data.

Proximity-based Approaches

Proximity based methods

- Proximity-based methods identify anomalies as those instances that are most **distant** from other objects.
- This relies on the **assumption** that **normal instances** are **related** and hence **appear close** to each other, while **anomalies** are **different** from the other instances and hence are **relatively far** from other instances.
- Proximity based approaches are known as model-free anomaly detection methods, since they do not construct a model of the normal class for computing the anomaly score.

Distance-based Anomaly Score

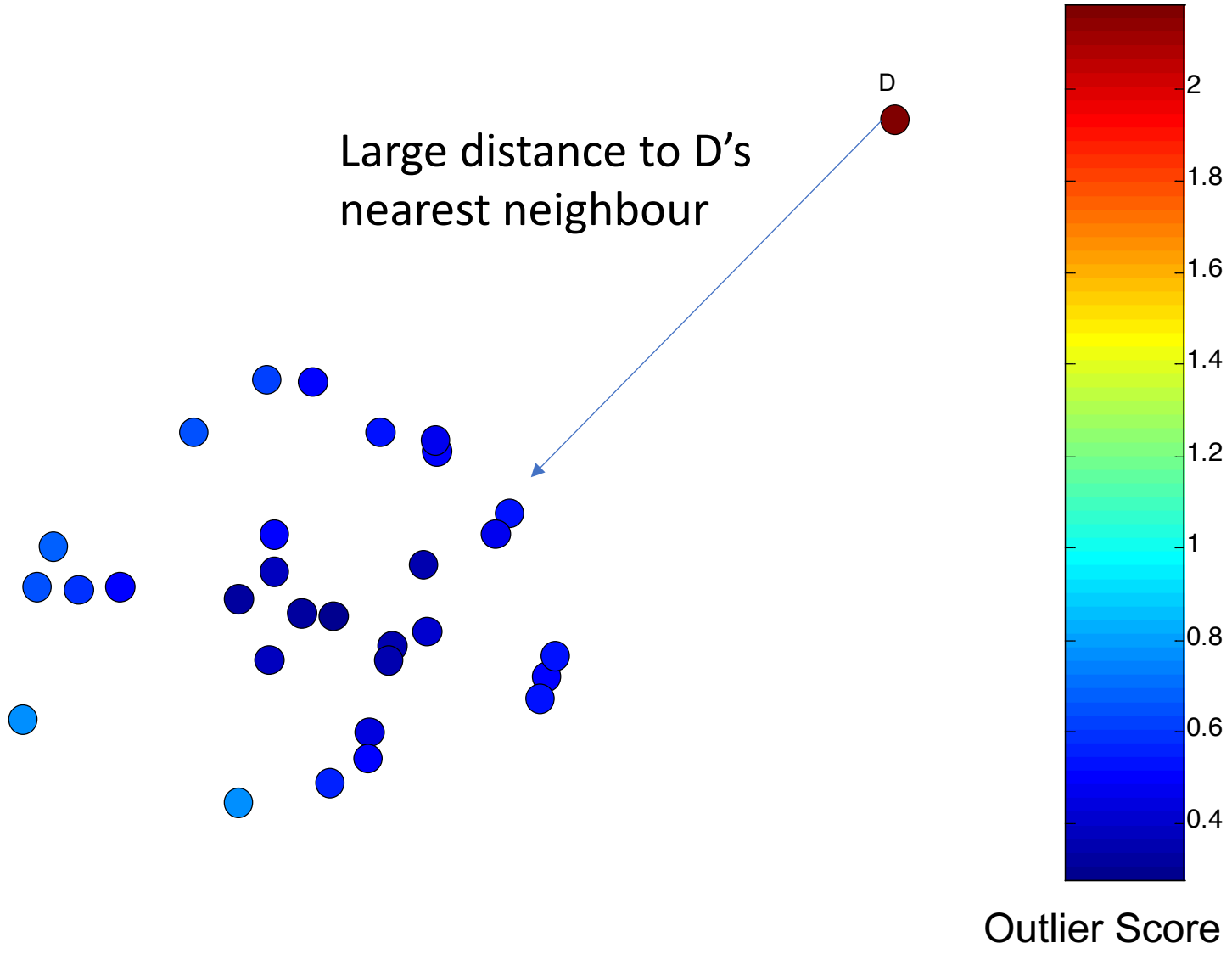
- One way to assign a **proximity based anomaly score** of a data instance x is to use the distance to its k^{th} nearest neighbour, $dist(x, k)$.
- If an instance x has many other instances located close to it (characteristics of the normal class), it will have a **low** value of $dist(x, k)$.
- An anomalous instance x will be quite distant from its k-neighbouring instances and would thus have a **high** value of $dist(x, k)$.
- Choosing k is significant (illustrated in the following slides).

One Nearest Neighbour - One Outlier

$k = 1$

Correctly identifies the single outlier D

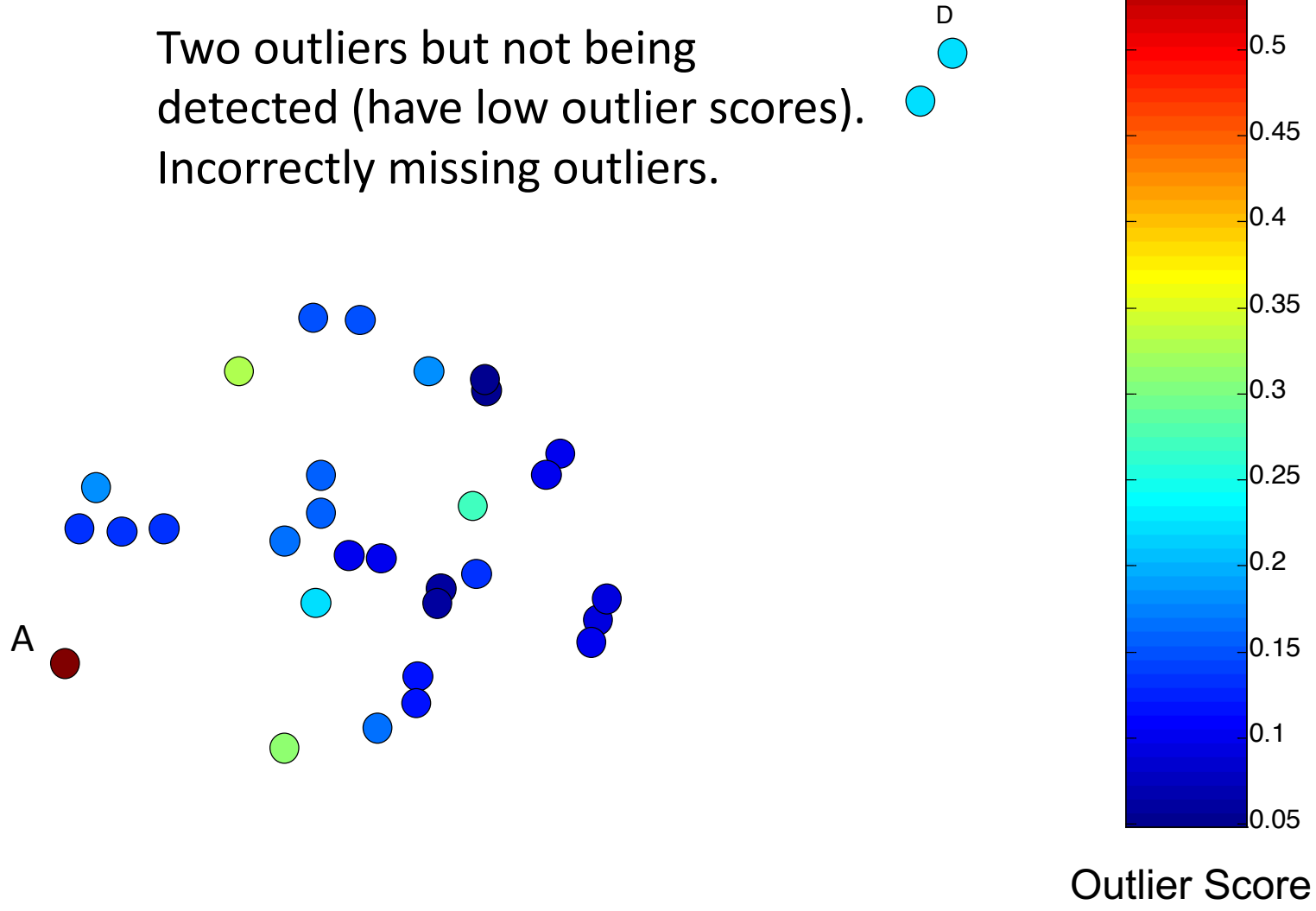
Large distance to D's nearest neighbour



One Nearest Neighbour - Two Outliers

$k = 1$

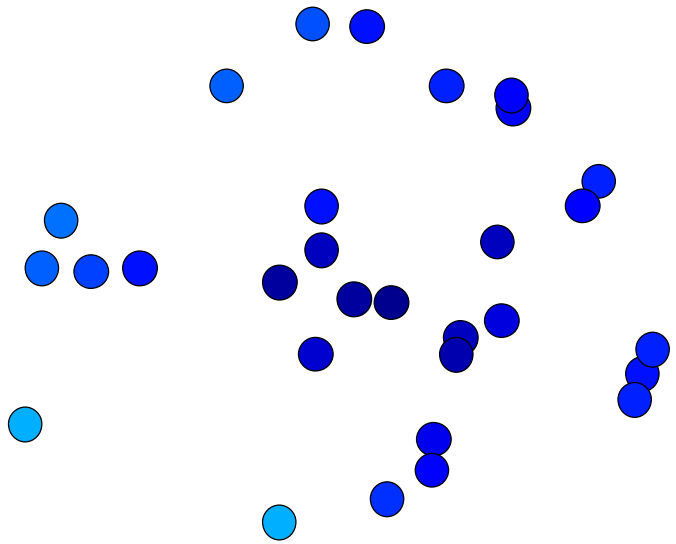
Two outliers but not being detected (have low outlier scores). Incorrectly missing outliers.

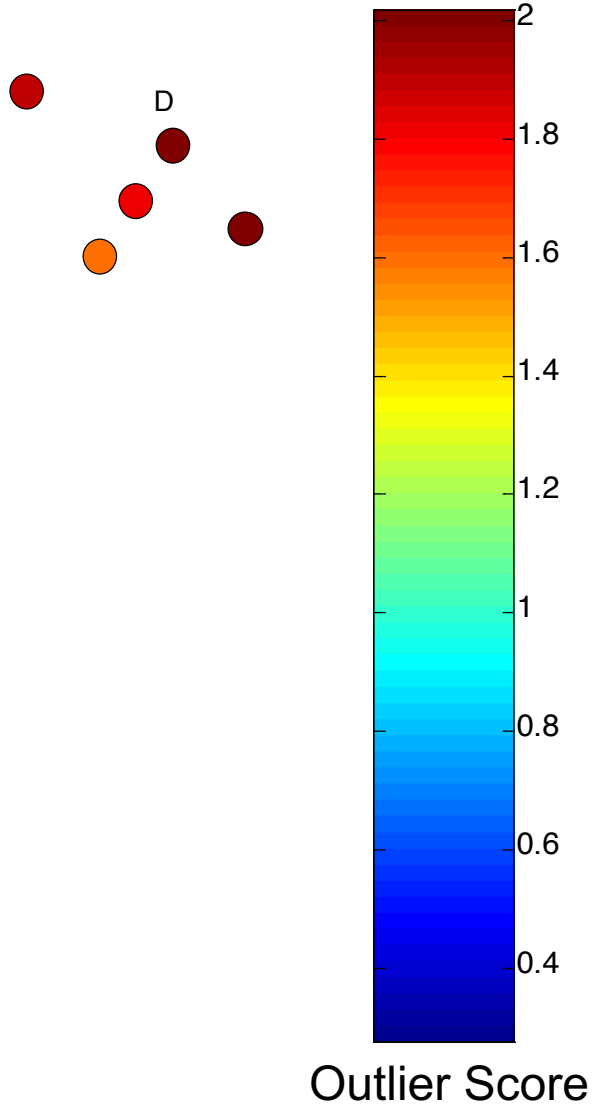


Five Nearest Neighbours - Small Cluster

$k = 5$

Small cluster around D but each is identified as an outlier because the distance to the 5th nearest neighbour is large.
Incorrectly identifying instances in small cluster as outliers.





Average distance as a proximity anomaly score

An alternative distance-based anomaly score that is to use the average distance rather than the distance to the k^{th} nearest neighbour.

average.dist(x,k) is the average distance of the k nearest neighbours.

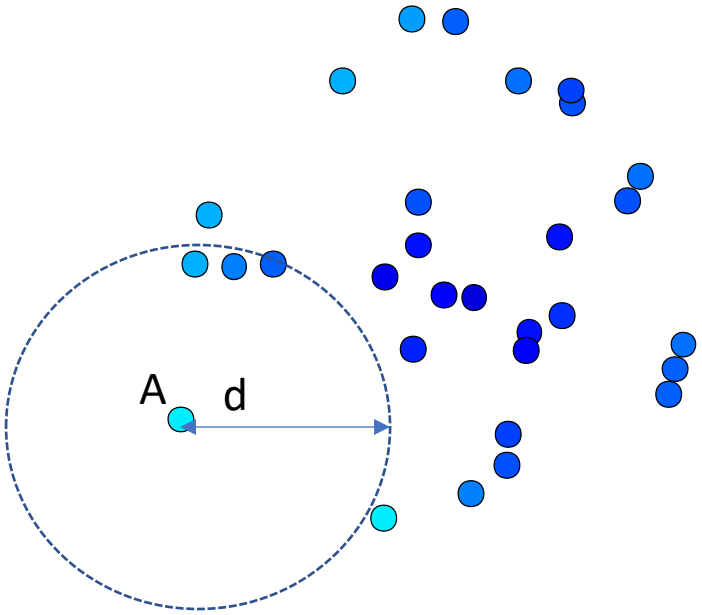
This is more widely used as a reliable proximity based anomaly score.

Density based approach

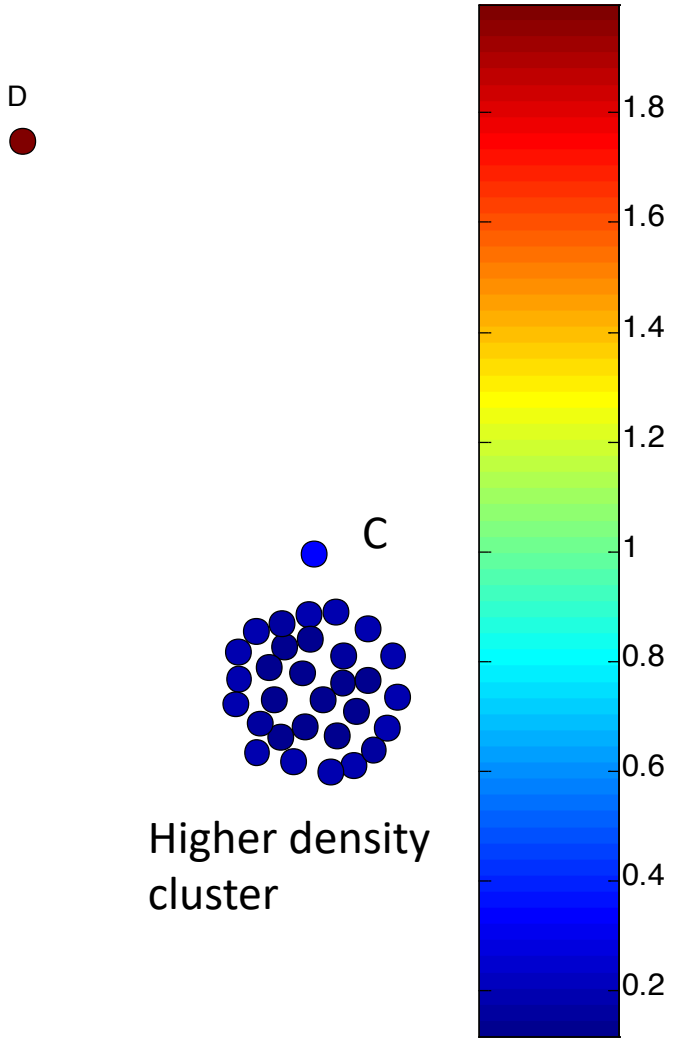
- The density around an instance can be defined as $n/V(d)$, where n is the number of instances within a specified distance d from the instance and $V(d)$ is the volume of the neighbourhood. Since $V(d)$ is constant for a given d, the density around an instance is represented using the number of instances n within a fixed distance d.
- The anomalies are the instances which are in low density.
- It is challenging to choose the parameter d in density based measures. If d is **too small** then many normal instances can incorrectly show low density values. If d is **too large**, then many anomalies may have densities that are similar to normal instances.

Density based methods

If d is small then normal instances in a low density cluster may be classified as outliers (incorrectly classified)



Low density cluster

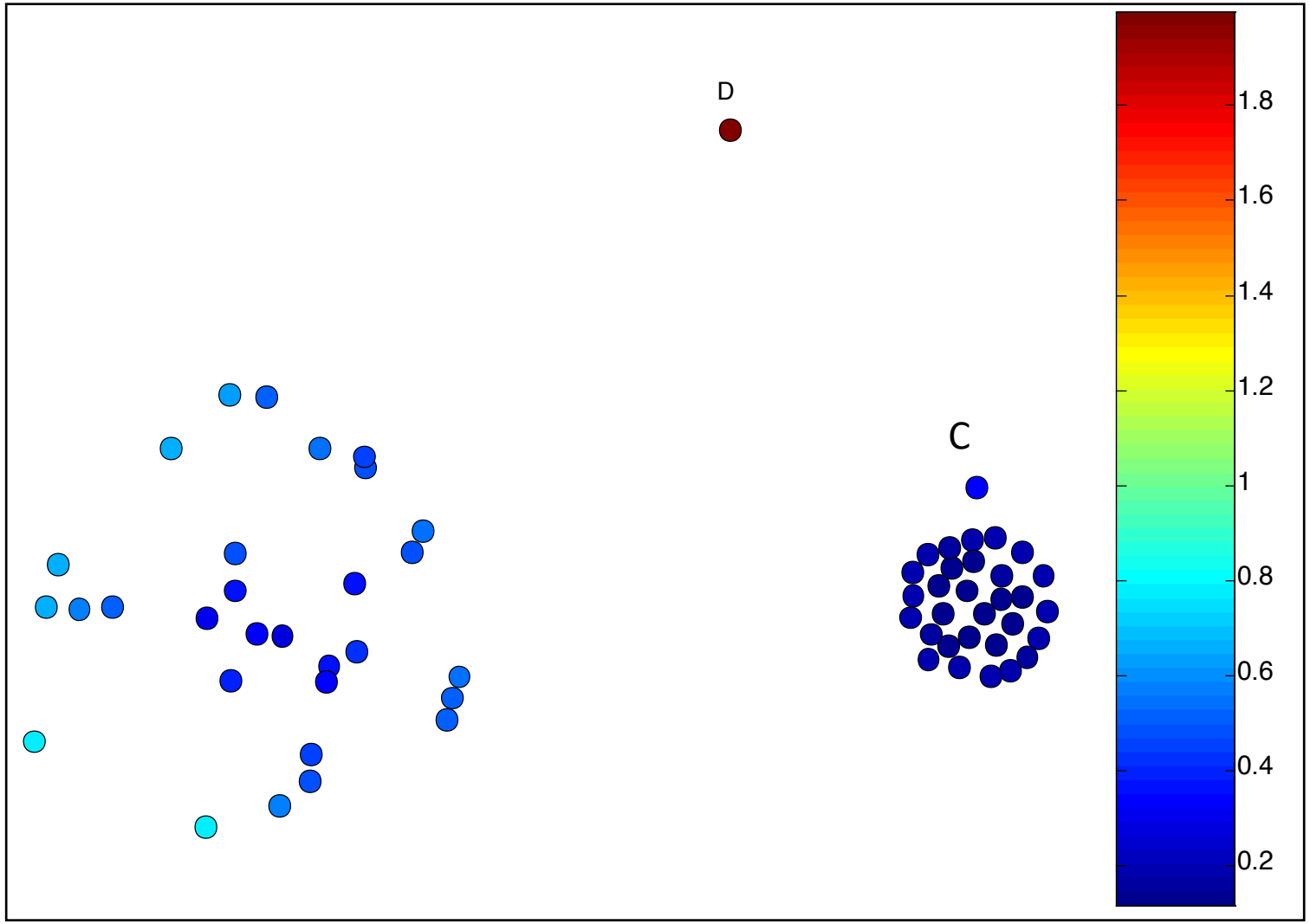


Higher density cluster

Outlier Score

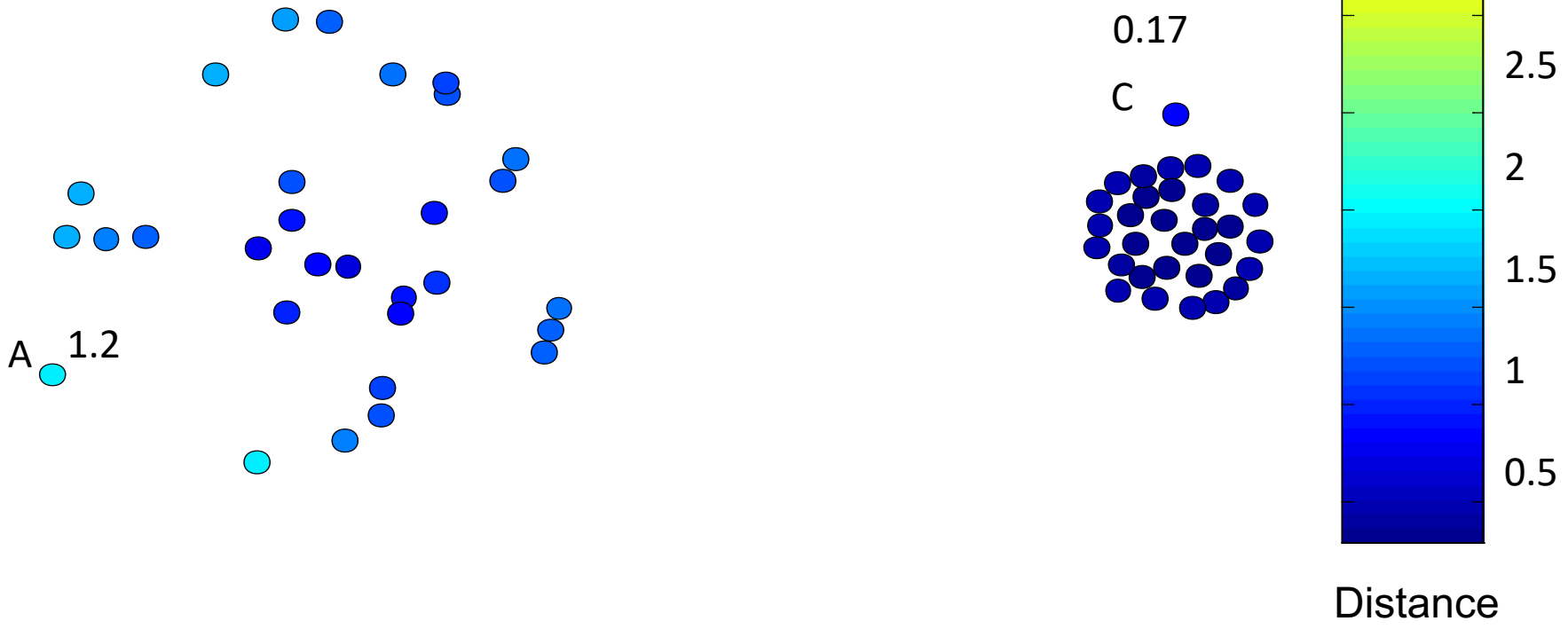
Relative Density-based Anomaly Score

- In situations where there are regions of high and low density the notion of a normal instance varies across these regions.
- For example C has a low k^{th} nearest neighbour score so it would not be identified as an anomaly but it is anomalous to the high density cluster. So we need to identify the notion of relative density.



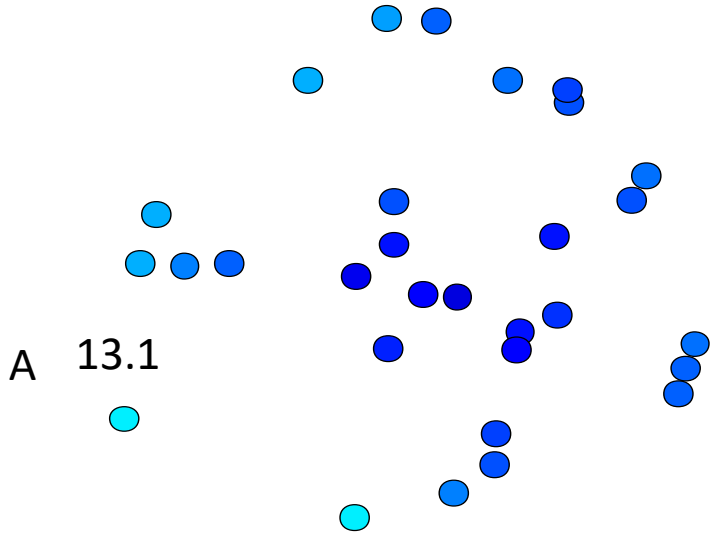
Example

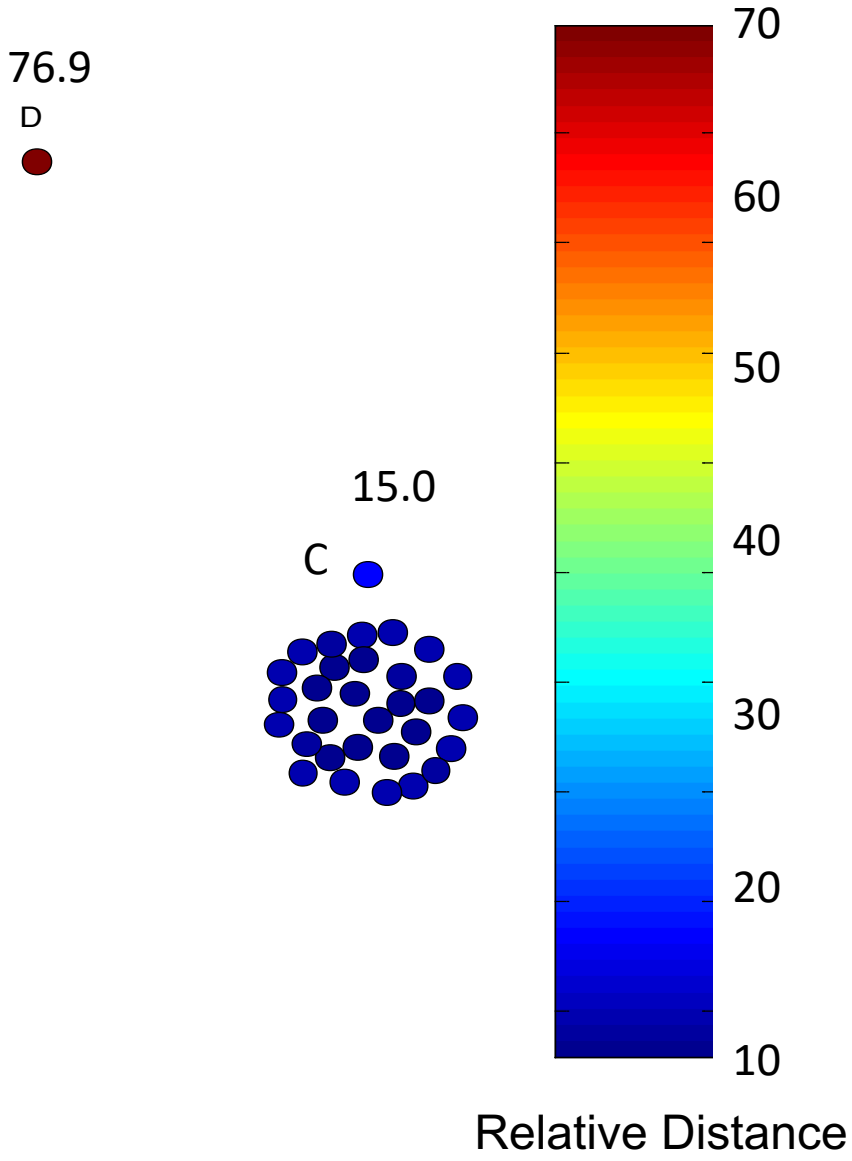
The distances from the points
to its nearest cluster centroid.
C has a small distance but seems to
be an anomaly whereas A has a larger score
but its not an anomaly because of the loose cluster.



Example

A more sophisticated approach is to use the relative distances. This is the ratio of the distance from the point to its centroid divided by the median distance of the points in the cluster from the centroid.



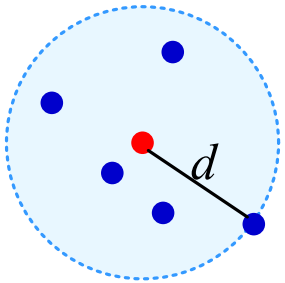


Local Outlier Factor

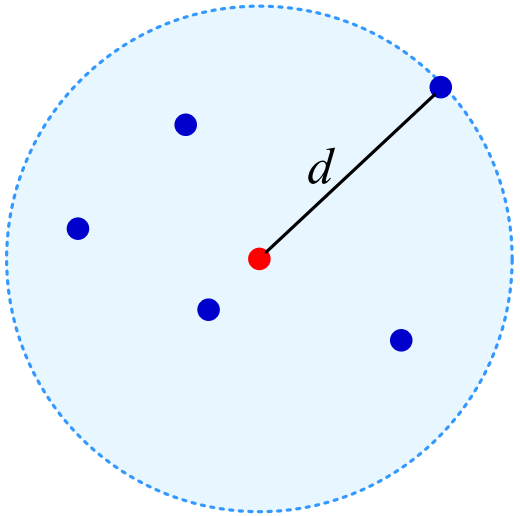
Let x be a data point. Fix a value of k. (In the diagrams we will take $k = 5$.)

Let $d = dist(x, k)$ be the distance from x to its k^{th} nearest neighbour.

Then d measures the size of the domain containing the k nearest neighbours of x.

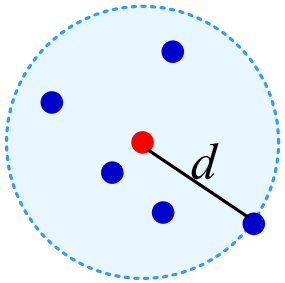


More densely packed points

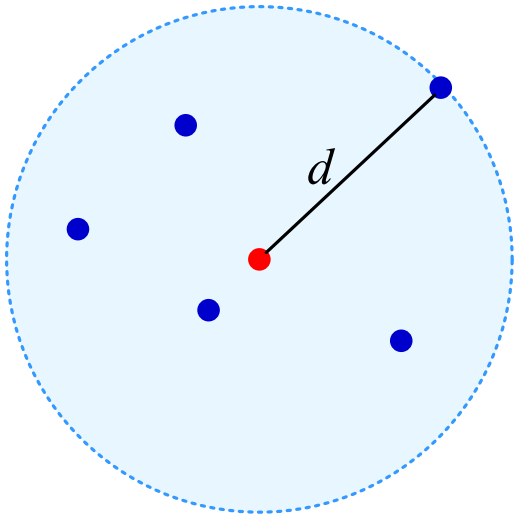


Less densely packed points

Local Outlier Factor



More densely packed points



Less densely packed points

Then d will be smaller in more densely-packed regions and will be larger in less densely-packed regions.

So $1/d = 1/dist(x,k)$ represents the density of the region around x.

We define $density(x,k) = 1/dist(x,k)$ as the density of x.

Local Outlier Factor

Then $density(x, k)$ is an anomaly measure for x :

- normal points will have higher density scores
- anomalies will have lower density scores.

However the ‘pure’ density score is not a good anomaly measure if there are regions of different densities in the data.

Local Outlier Factor

The local outlier factor measures the relative density of a point x compared with the densities of its neighbouring points.

Let $x_1, x_2, \ldots, x_k$ denote the k nearest neighbours of x.

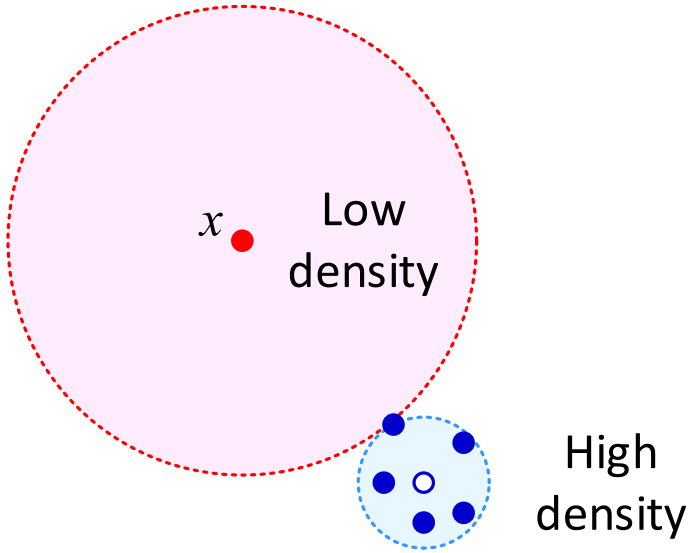
Then the local outlier factor is

$$LOF(x) = \frac{average\ density\ of\ x_1, \ldots, x_k}{density\ of\ x} = \frac{\frac{1}{k}\sum_{i=1}^{k} density(x_i, k)}{density(x, k)}$$

An anomaly will have lower density compared with its neighbours.

Local Outlier Factor

An anomaly will have lower density compared with its neighbours.



For an anomaly, $LOF(x) \gg 1$.

For a normal point, $LOF(x) \approx 1$

Strengths and Weaknesses

- Proximity-based approaches are non-parametric in nature and hence are not restricted to any particular form of distribution of the normal and anomalous classes.
- They have a broad applicability over a wide range of anomaly detection problems where a reasonable proximity measure can be defined between instances.
- They are quite intuitive and visually appealing, since proximity-based anomalies can be interpreted visually when the data can be displayed in two- or three-dimensional scatter plots.

Strengths and Weaknesses

- However, the effectiveness of proximity-based methods depends greatly on the choice of the distance measure.
- Defining distances in high-dimensional spaces can be challenging.
  - In some cases, **dimensionality reduction techniques** can be used to map the instances into a **lower dimensional feature space**.
- Another challenge common to all proximity-based methods is their high computational complexity.
- Choosing the right value of parameters (*k* or *d*) in proximity-based methods is also difficult and often requires domain expertise.

Clustering-based Approaches

Finding anomalous clusters

- This approach assumes the presence of clustered anomalies in the data where the anomalies appear in tight groups of small size.

- Clusters of anomalies are generally small in size and are expected to be away from the normal class since anomalies do not conform to normal patterns or behaviours.

Clustering based approaches

Clustering-based methods can be categorised into two types:

a) Methods that consider **small clusters** as anomalies, and

b) Methods that define a **point as anomalous** if does not fit the clustering well, typically as measured by distance from a cluster centre.

Finding anomalous instances

- An approach is to cluster all instances and then assess the degree to which each instance belongs to its respective cluster.
- For example, if k-means clustering is used the distance to its cluster centroid represents how strongly the instance belongs to the cluster.
- Instances that are relatively distant from their cluster centroid can be identified as anomalies.

Finding anomalous instances

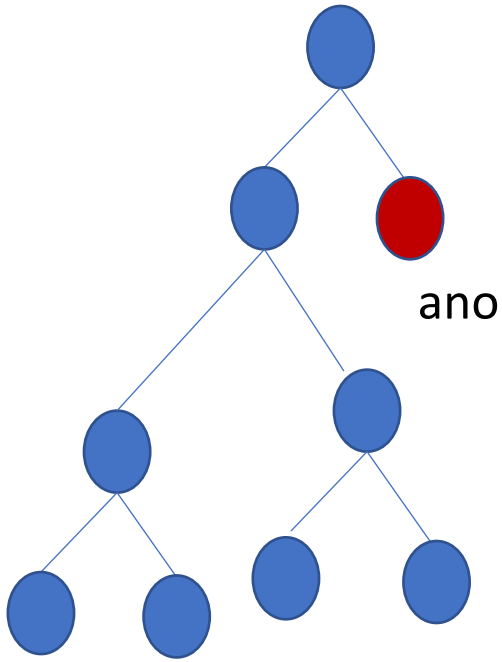
- There are several methods which can be used to assess whether an instance belongs to a cluster.
- One method is to measure the distance of the instance from the cluster prototype and consider this as its anomaly score of the instance.
- If the clusters are of differing densities we can construct an anomaly score that measures the **relative** distance of the instance from the prototype with respect to other instances in the cluster.

Ensemble Methods

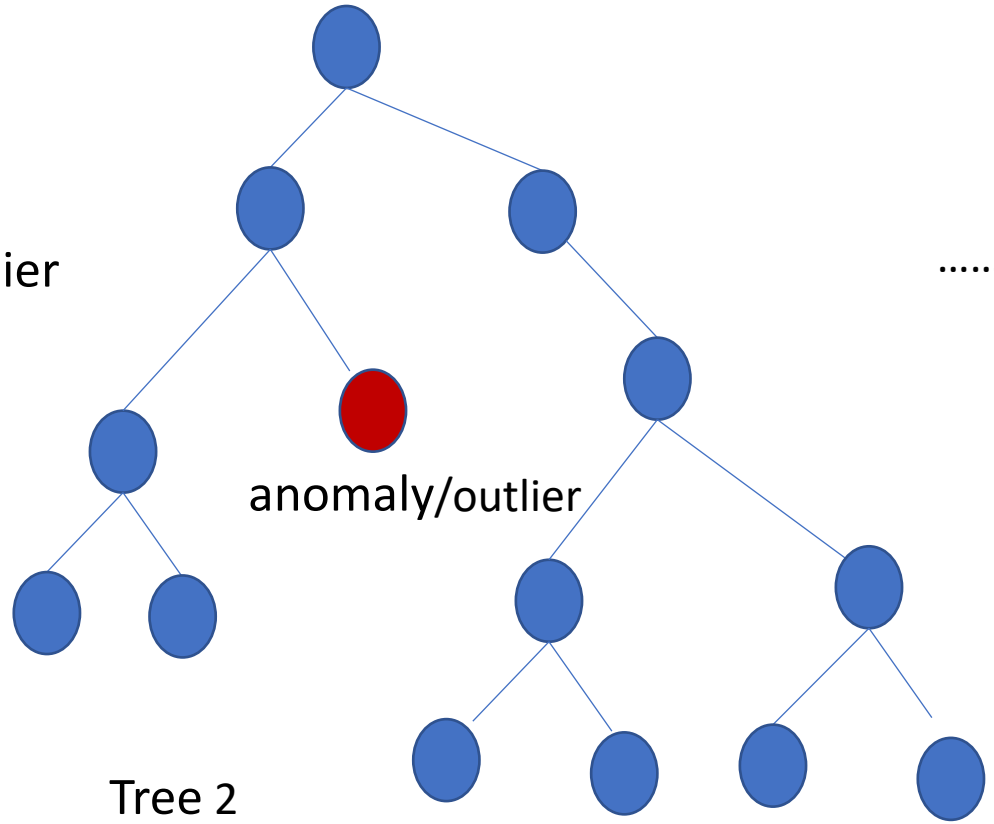
Isolation Forest

- Unsupervised machine learning technique for anomaly detection.
- Isolation Forest is an Ensemble method similar to Random Forest.
- The algorithm takes a feature at random and builds a decision tree
- Multiple trees are created through this process.
- The intuition here is that anomalies are isolated at an earlier stage of the construction of the decision tree (earlier in terms of depth).
- An anomaly score is given to each observation.

Example

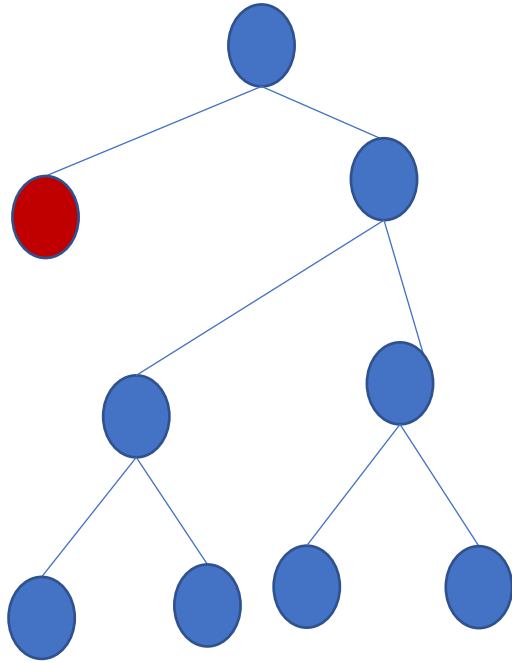


anomaly/outlier



anomaly/outlier

.....



Tree 1

Tree 2

Tree n

Anomaly Score

The anomaly score is determined using the equation

$$S(x, m) = 2^{-E(h(x))/c(m)}$$

where:

- x is a particular data point and m is the number of points.
- $E\big(h(x)\big)$ is the average path length to data point x across all of the tree.
- $c(m)$ is the average depth of data points across all the trees.

Intuition of the anomaly score

If $E\big(h(x)\big) \ll c(m)$ then $S(x,m) \approx 1$ and x is an anomaly.

This is observed when $E\big(h(x)\big)$ is very small ($2^{-small\ number} \approx 1$).

Hence the average depth of the data point is higher up in the tree.

If $E\big(h(x)\big) \gg c(m)$ then $S(x,m) \approx 0$ and x is a normal data point.

This is observed when $E\big(h(x)\big)$ is large ($2^{-high\ number} \approx 0$).

Hence the depth of the data point is further down the tree.

Isolation forest in Python

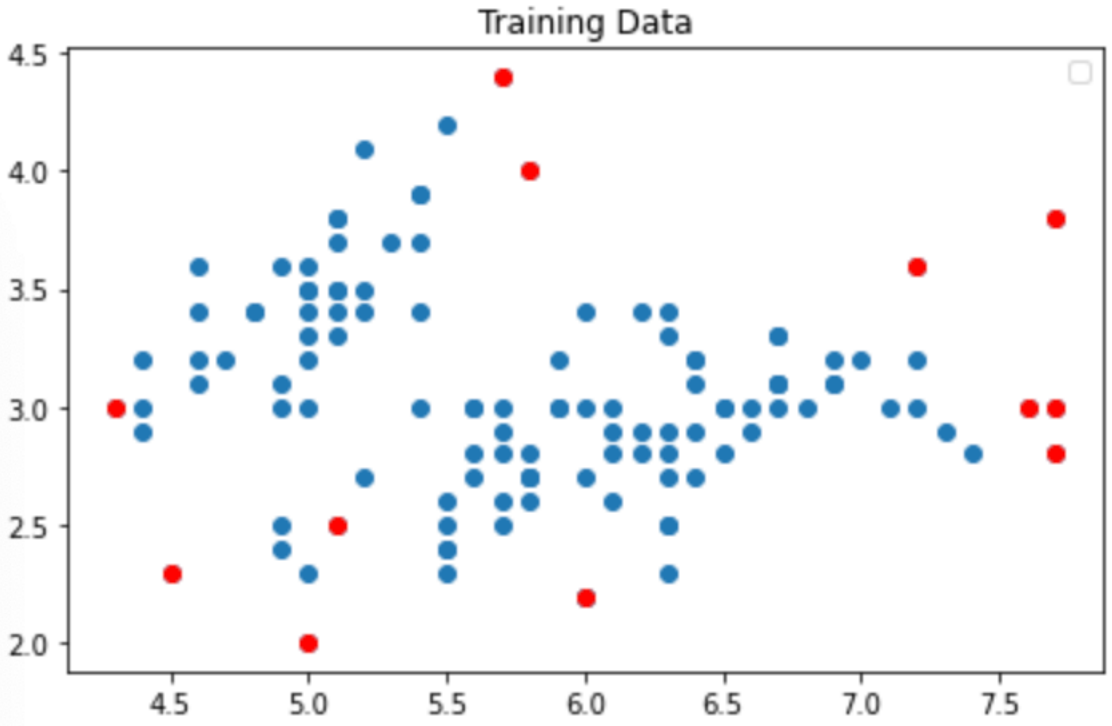
```
from sklearn.ensemble import IsolationForest

model=IsolationForest(n_estimators=50,contamination=0.1)
```

n_estimators – total number of trees to use, the default value is 100

Contamination- the expected proportion of outliers in the data

```
model.fit(df[[X_train]])
```

```
y_pred_train = model.predict(X_train)
X_train['outlier']=y_pred_train
X_=X_train[X_train['outlier']<1]
```



	sepal_length	sepal_width	petal_length	petal_width	outlier
22	4.6	3.6	1.0	0.2	-1
15	5.7	4.4	1.5	0.4	-1
65	6.7	3.1	4.4	1.4	1
11	4.8	3.4	1.6	0.2	1
42	4.4	3.2	1.3	0.2	1
...	...	...	...	...	...
71	6.1	2.8	4.0	1.3	1

Reconstruction-Based Approaches

- Based on assumptions there are patterns in the distribution of the normal class that can be captured using lower-dimensional representations

- Reduce data to lower dimensional data
  - E.g. Use Principal Components Analysis (PCA) or Auto-encoders

- Measure the reconstruction error for each object
  - The difference between original and reduced dimensionality version

Reconstruction Error

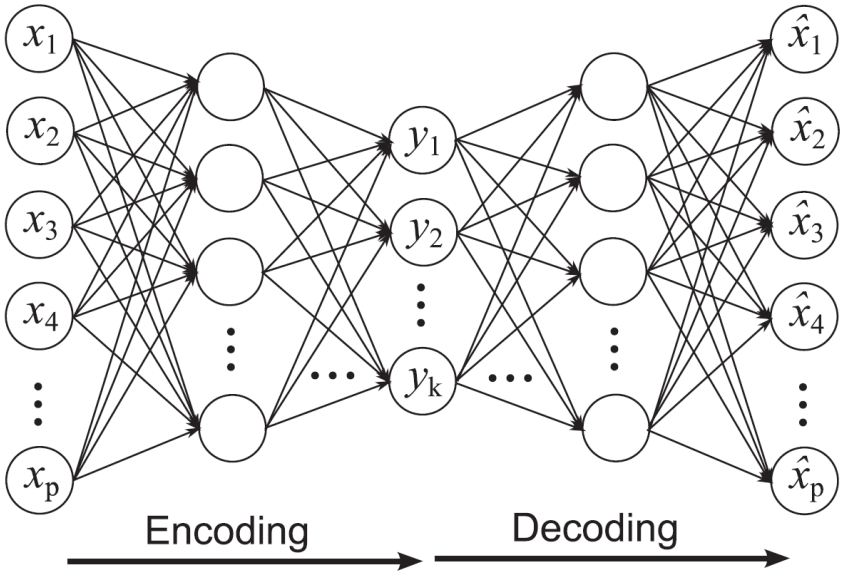
- Let $\mathbf{x}$ be the original data object
- Find the representation of the object in a lower dimensional space
- Project the object back to the original space
- Call this object $\hat{\mathbf{x}}$

$$\text{Reconstruction Error}(\mathbf{x}) = \|\mathbf{x} - \hat{\mathbf{x}}\|$$

- Objects with large reconstruction errors are anomalies

Basic Architecture of an Autoencoder

- An autoencoder is a multi-layer neural network
- The number of input and output neurons is equal to the number of original attributes.



Strengths and Weaknesses

- Does not require assumptions about distribution of normal class
- Can use many dimensionality reduction approaches
- The reconstruction error is computed in the original space
  - This can be a problem if dimensionality is high