Estimação, Deteção e Aprendizagem II

Deteção de Outliers

Catarina Oliveira

DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

UPT UNIVERSIDADE PORTUCALENSE

CONTEÚDO

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

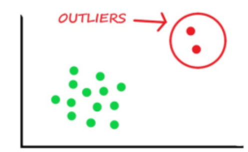
Outliers

Outlier: observação que se desvia significativamente do resto das observações (diferente de ruído)

Aplicações: deteção de fraude, medicina, segurança, indústria, processamento de imagem, vigilância de redes de sensores/vídeo, deteção de intrusões

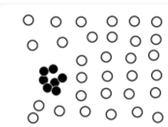
Tipos de *outliers*

Global:
- Observação que se desvia significativamente das outras
- Tipo mais simples de *outlier*
- Maioria dos métodos tem como objetivo detetar este tipo



Contextual / Condicional:
- Observação que se desvia significativamente das outras relativamente a um determinado contexto (ex: a temperatura hoje é 30ºC – em Dezembro é um *outlier*; em Julho não)
- O contexto tem de ser especificado juntamente com a definição do problema
- Atributos separados em dois tipos:
  - <u>Contextuais</u>: definem o contexto (ex: data, localização)
  - <u>Comportamentais</u>: definem as características do objeto, sendo usados para avaliar se é ou não um *outlier* no contexto a que pertence (ex: temperatura, humidade)

Coletivo:
- Conjunto de objetos que se desviam significativamente dos restantes
- Cada objeto isoladamente não é um *outlier*



DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Desafios da deteção de *outliers*

Modelação de objetos normais e *outliers* de forma eficaz

A fronteira entre a normalidade e a anormalidade dos dados (*outliers*) geralmente não é bem definida

Dependente dos dados

Ex: em medicina, uma variação pequena pode ser significativa; em marketing seria necessária uma variação grande para ser significativa

Lidar com o ruído

É necessário remover o ruído antes da deteção de *outliers* para evitar que o *outlier* seja "mascarado" pelo ruído

Compreensibilidade

Justificar o *outlier* detetado

Métodos de deteção de *outliers*

Supervised

Especialistas identificam dados como sendo normais / *outliers* e depois pode ser vista como um problema de classificação

Desafios:

- Classes desbalanceadas (dados normais são em muito maior quantidade que *outliers*)
- Encontrar tantos *outliers* quanto possível é mais importante do que não classificar erradamente normais como *outliers*

Unsupervised

Não sabemos que objetos são normais/*outliers*

Assume-se que os objetos estão “*clustered*” e que os *outliers* estão mais longe

Semi-supervised

Semelhante ao *supervised*, mas com apenas um subconjunto dos dados identificados como normais/*outliers*

Métodos de deteção de *outliers*

Estatísticos (*model-based*)

Assumem: dados gerados por um modelo estatístico (estocástico); dados que não seguem o modelo são outliers

Proximidade

Assumem: um objeto é um *outlier* se os seus vizinhos mais próximos estão distantes do *feature space* (i.e: a proximidade do objeto aos seus vizinhos desvia-se da proximidade da maioria dos objetos aos seus vizinhos)

Clustering

Assumem: objetos normais pertencem a *clusters* densos e de grande dimensão e *outliers* pertencem a *clusters* pequenos ou esparsos ou não pertencem a nenhum *cluster*

Métodos estatísticos

Métodos estatísticos

Assumem que os objetos normais num conjunto de dados são gerado por um processo estocástico:

- Objetos normais ocorrem em regiões de alta probabilidade para o modelo estocástico
- Objetos nas regiões de baixa probabilidade são *outliers*

Duas categorias:

- **Paramétricos**: assumem que os objetos normais são gerados por uma distribuição paramétrica com parâmetro Θ. A *probability density function* da distribuição paramétrica $f(x, \Theta)$ determina a probabilidade de x ser gerado por essa distribuição. Quanto menor esse valor, x tem uma maior probabilidade de ser um *outlier*
- **Não-paramétricos**: não assumem nenhum modelo estatístico a priori, mas tenta determinar o modelo a partir dos dados de input

Métodos estatísticos paramétricos

Dados *univariate*: assumir distribuição normal – usar *maximum likelihood*

Exemplo:

Considerando uma amostra de *n* valores ordenados (ex: 24,0; 28,9; 28,9; 29,0; 29,1; 29,1; 29,2; 29,2; 29,3; 29,4)

Assumindo que os valores seguem a distribuição normal com média µ e desvio padrão σ

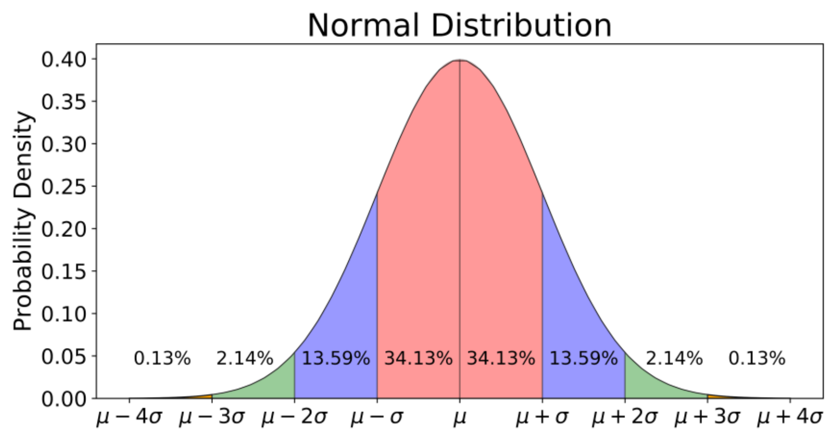
Obtém-se os *maximum likelihood estimates*:

$$\hat{\mu} = \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

No exemplo = 28,61

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

No exemplo ≈ 2,29 ⇒ $\hat{\sigma} \approx \sqrt{2{,}29} \approx 1{,}51$



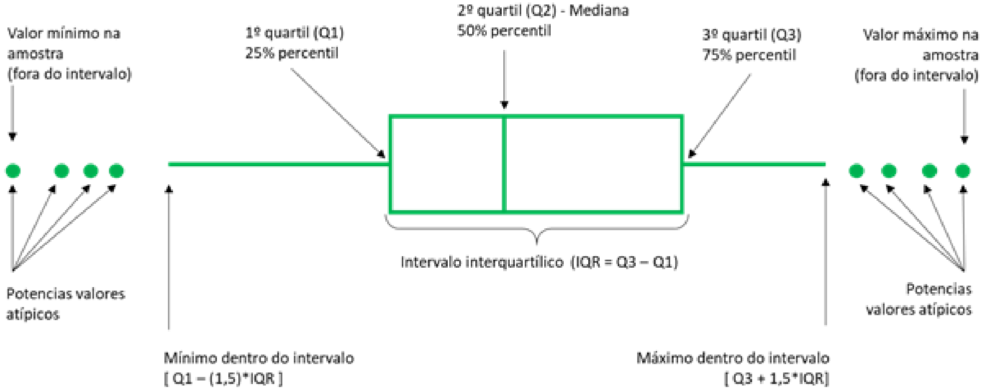
Na distribuição normal, $\mu \pm 3\sigma$ contém aproximadamente 97.7% dos dados

Um valor fora desse intervalo será, provavelmente, um *outlier*. Ou seja, os valores $v : \frac{\mu - v}{\sigma} > 3$, porque a probabilidade de seguirem a mesma distribuição é < 0,15%

No exemplo, 24,0 é um *outlier*, porque $\frac{28{,}61 - 24{,}0}{1{,}51} \approx 3{,}05 > 3$

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Dados *univariate*: assumir distribuição normal – usar *boxplot*



Outliers são os valores v :

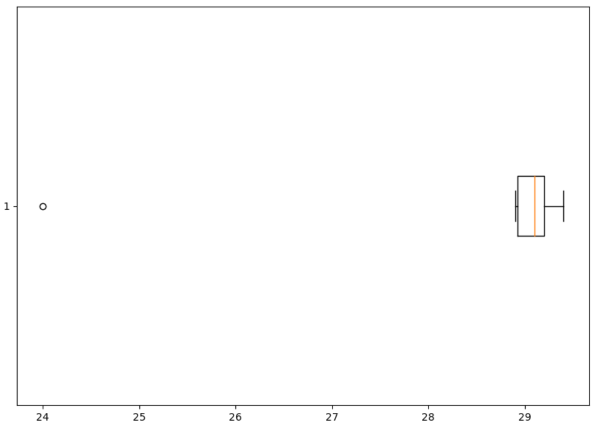
$v < Q1 - 1{,}5 \times IQR$

ou

$v > Q3 + 1{,}5 \times IQR$

Exemplo:

Considerando uma amostra de *n* valores ordenados (ex: 24,0; 28,9; 28,9; 29,0; 29,1; 29,1; 29,2; 29,2; 29,3; 29,4)




```
import matplotlib.pyplot as plt
fig = plt.figure(figsize =(10, 7))
plt.boxplot(x=[24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4], vert=False)
plt.show()
```

24,0 é um outlier

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Dados *univariate*: assumir distribuição normal – usar teste de *Grubb*

Para cada objeto x num conjunto de N valores com média $\bar{x}$ e desvio padrão s, definimos o seu z-score:

$$z = \frac{|x - \bar{x}|}{s}$$

O objeto x é um *outlier* se:

$z \geq \frac{N-1}{\sqrt{N}} \sqrt{\frac{t^2{}_{\alpha/(2N),N-2}}{N-2+t^2{}_{\alpha/(2N),N-2}}}$, em que:

$t^2{}_{\alpha/(2N),N-2}$ é o valor seguindo uma distribuição t com um nível de significância $\alpha/(2N)$ com $N-2$ graus de liberdade

Dados *multivariate*: assumir distribuição normal, transformar em *univariate* – usar distância de *Mahalanobis*

Seja $\bar{o}$ o vetor médio de um *dataset* D e S a matriz de covariância.

Para cada objeto o do *dataset*, a distância de *Mahalanobis* de o a $\bar{o}$ é:

$$MDist(o, \bar{o}) = (o - \bar{o})^{T} S^{-1}(o - \bar{o})$$

$MDist(o, \bar{o})$ é uma variável *univariate*, podendo aplicar-se o teste de *Grubb*.

Podemos transformar a deteção de *outliers* em dados *multivariate* da seguinte forma:

1. Calcular o vetor médio do *dataset*
2. Para cada objeto o calcular $MDist(o, \bar{o})$
3. Detetar *outliers* no *dataset univariate* transformado $\{MDist(o, \bar{o}), o \in< D\}$
4. Se se determinar que $MDist(o, \bar{o})$ é um *outlier*, então o também é um *outlier*

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Dados *multivariate*: assumir distribuição normal, transformar em *univariate* – usar Qui quadrado

Para cada objeto o de um *dataset* com n observações, o Qui quadrado é:

$$\chi^2 = \sum_{i=1}^{n} \frac{(o_i - E_i)^2}{E_i}$$

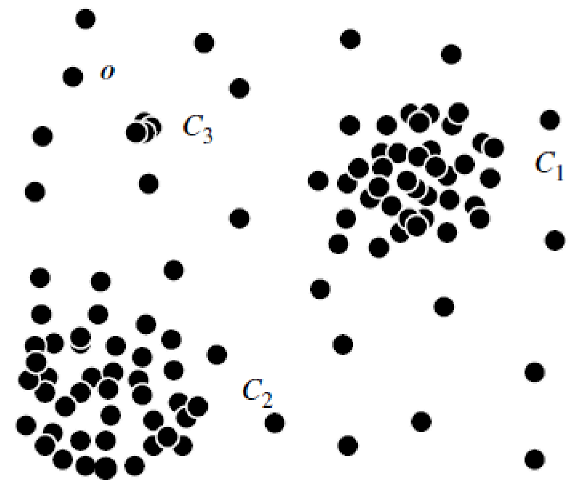
, o_i é o valor de o na i-ésima dimensão; E_i é a média da i-ésima dimensão

Se o valor do χ^2 for alto, o objeto o é um *outlier*.

Dados *multivariate*: assumir múltiplas distribuições normais

Considerando os dados da figura, há dois clusters.

Assumir que os dados são gerados por uma distribuição normal iria estimar a média a meio dos dois clusters, e os objetos entre os clusters não iriam ser detetados como *outliers*.



Podemos assumir que os objetos normais são gerados por diversas distribuições normais.

Neste caso, com 2 distribuições, assumimos as distribuições normais:

$\Theta_1(\mu_1, \sigma_1)$ e $\Theta_2(\mu_2, \sigma_2)$

Para cada objeto *o* do *dataset*, a probabilidade de *o* ser gerado pela composição das duas distribuições é:

$\Pr(o|\Theta_1, \Theta_2) = f_{\Theta_1}(o) + f_{\Theta_2}(o)$, f_{Θ_1} e f_{Θ_3} são as *probability density functions* de Θ_1 e Θ_2, respetivamente.

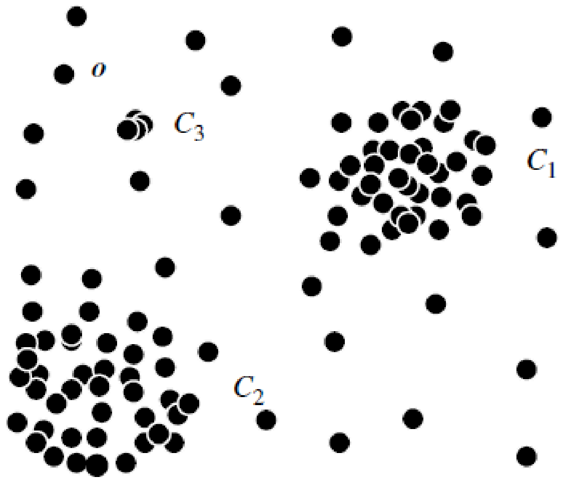
Podemos usar o algoritmo *Expectation Maximization* (EM) [(1)] para obter os parâmetros $\mu_1, \sigma_1,\ \mu_2\ e\ \sigma_2$.

Um objeto *o* é um *outlier* se não pertencer a nenhum cluster

(1) https://scikit-learn.org/stable/modules/mixture.html

Dados *multivariate*: usar múltiplos clusters

Cluster C3 deve ser detetado como *outlier*



Podemos assumir que os objetos normais são gerados por uma distribuição normal, ou uma composição de distribuições normais, e que os *outlliers* são gerados por outra distribuição.

Por exemplo, podemos assumir que esta distribuição tem uma maior variância se os *outliers* estiverem distribuídos numa área maior.

Na prática, definimos $\sigma_{outlier} = k\sigma$, em que *k* é um parâmetro definido pelo utilizador e σ é o desvio padrão da distribuição normal que gera os dados.

Podemos também usar o algoritmo EM

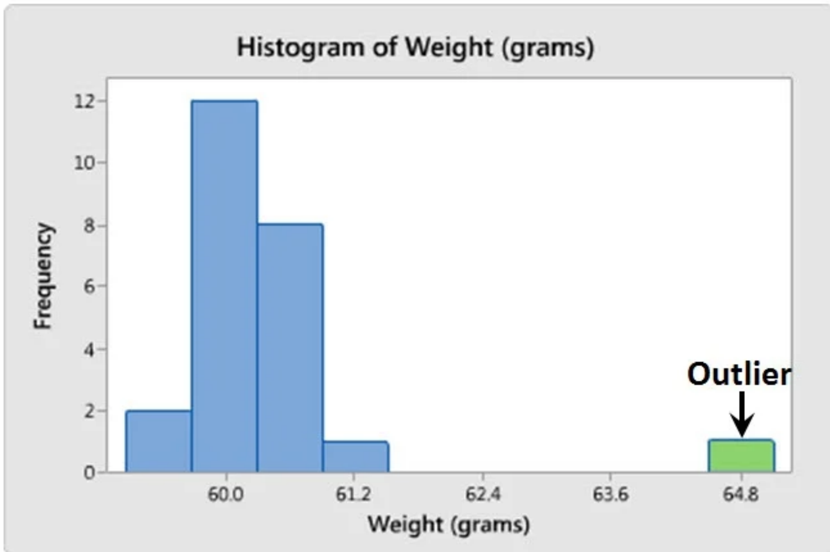
Métodos estatísticos não paramétricos

Histograma

Procedimento:

1. Construir o histograma a partir dos dados
2. Determinar os *outliers*: objetos que pertencem às “*bins*” menos populadas ou às mais “longe”

Exemplo:



Métodos de proximidade

Métodos de proximidade

Dado um conjunto de objetos num *feature space*, pode usar-se uma medida de distância para quantificar a semelhança entre objetos. Objetos mais longe podem ser considerados *outliers*.

Assumem: a proximidade de um *outlier* aos seus vizinhos mais próximos desvia-se significativamente da proximidade do objeto à maior parte dos outros objetos do conjunto

Dois tipos de métodos:

- ***Distance-based***: consulta a vizinhança de um objeto, definida por um determinado raio. Um objeto é considerado *outlier* se a sua vizinhança não tem pontos suficientes
    - *Outliers* globais (tendo em conta todo o dataset)
- ***Density-based***: investiga a densidade de um objeto e a dos seus vizinhos. Um objeto é um *outlier* se a sua densidade é muito inferior à dos seus vizinhos.
    - Permite *outliers* locais (tendo em conta vizinhanças locais)

Métodos de proximidade

Distância

Deteção de *outliers* por proximidade - distância

Num *dataset D* de objetos, define-se um *threshold* de distância, *r,* para a vizinhança de um objeto

Para cada objeto, *o*, verifica-se o número de objetos na sua *r*-vizinhança

Se a maioria dos objetos de *D* estiverem longe de *o* (não estiverem na sua *r*-vizinhança), então *o* é um *outlier*

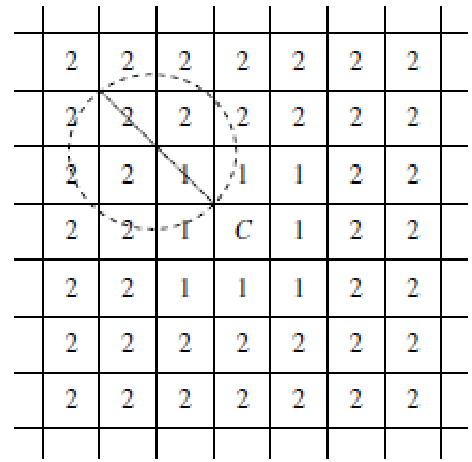
Seja π $(0 < \pi < 1)$ um *threshold* (fração). Um objeto *o* é um $DB(r, \pi)$-*outlier* se:

$$\frac{\|\{o' | dist(o, o') \leq r\}\|}{\|D\|} \leq \pi$$

(dentro da vizinhança de raio r há menos do que π objetos)

Deteção de *outliers* por proximidade – distância – grelha (método CELL)

Feature space é particionado numa grelha multidimensional, em que cada célula é um "hipercubo" com diagonal de tamanho $\frac{r}{2}$, em que *r* é o *threshold* de distância. Se o dataset tiver *l* dimensões, o tamanho da aresta de cada célula será $\frac{r}{2\sqrt{l}}$



Considerando um dataset 2D, o comprimento da aresta de cada célula é $\frac{r}{2\sqrt{2}}$

A célula C tem a elementos; b_1 e b_2 são o número total de elementos das células marcadas com 1 e 2, respetivamente

As células vizinhas de C podem ser divididas em 2 grupos de diferentes níveis:

- **Nível 1** - contíguas a C
    - Dado qualquer ponto $x \epsilon C$ e qualquer ponto possível y numa célula de nível 1, então $dist(x, y) \leq r$
    - Se $a + b_1 > \lceil \pi n \rceil$, todos os objetos *o* de C <u>não são</u> $DB(r, \pi)$-*outliers*, porque todos os objetos de C e das células nível 1 estão na *r*-vizinhança de *o* e há pelo menos $\lceil \pi n \rceil$ vizinhos com essas características
- **Nível 2** - à distância de 1 ou 2 células de C
    - Dado qualquer ponto $x \epsilon C$ e qualquer ponto possível y tal que $dist(x, y) \geq r$, então y está numa célula de nível 2
    - Se $a + b_1 + b_2 < \lceil \pi n \rceil + 1$, todos os objetos de C <u>são</u> $DB(r, \pi)$-*outliers*, porque cada uma das suas *r*-vizinhanças tem menos do que $\lceil \pi n \rceil$ objetos

Métodos de proximidade

Densidade

Deteção de *outliers* por proximidade – densidade – proximidade local

Assume: densidade relativa (à volta) de um objeto normal é significativamente diferente da densidade relativa dos seus vizinhos

Dado um objeto *o* e um conjunto de objetos *D*, a distância-*k*, $dist_k(o)$, é a distância $\mathrm{dist}(o, p)$ entre os objetos *o* e *p* tal que:

- Há pelo menos *k* objetos $o' \in D - \{o\}$ tal que $\mathrm{dist}(o, o') \leq dist(o, p)$
- Há no máximo *k-1* objetos $o'' \in D - \{o\}$ tal que $\mathrm{dist}(o, o'') < dist(o, p)$

Ou seja, $dist_k(o)$ é a distância entre *o* e os seus *k* vizinhos mais próximos

A *k-distance-neighborhood* de *o* contém todos os objetos cuja distância para *o* não é maior do que $dist_k(o)$.

A densidade local de *o* é pode ser a média das distâncias de o aos objetos na *k-distance-neighborhood* de *o*

Clustering

Deteção de *outliers* com *clustering*

Após executar o *clustering*, vamos verificar quais são os *outliers*.

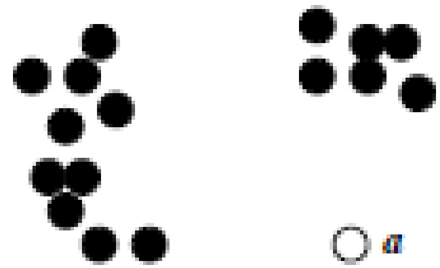
Outliers:

- Objetos que não pertencem a nenhum cluster
- Objetos que estão longe do cluster mais próximo
- Objetos que fazem parte de um cluster pequeno ou esparso

Deteção de *outliers* com *clustering*: objetos que não pertencem a nenhum *cluster*

Usando um algoritmo de *clustering density-based*, (ex: DBSCAN) conseguimos determinar que:

- Os pontos pretos pertencem a clusters
- O ponto branco *a* não pertence a nenhum cluster
    - É um outlier



Deteção de *outliers* com *clustering*: objetos longe do *cluster* mais próximo

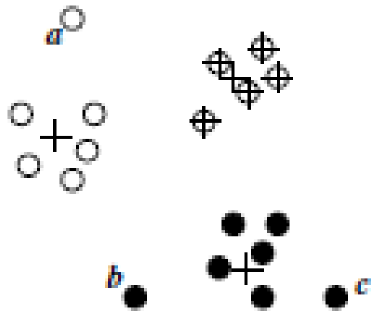
Usando, por exemplo, o *k-means* conseguimos particionar os dados em 3 clusters

diferentes símbolos

O centro de cada cluster está marcado com +

A cada objeto *o* podemos atribuir um score de acordo com a distância entre o objeto e o centroide mais próximo e comparar essa distância com a dos outros elementos do cluster

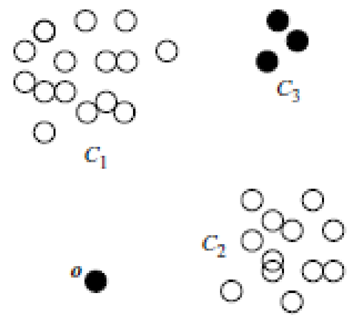
Se houver uma diferença muito grande, o objeto é um *outlier*



Deteção de *outliers* com *clustering*: objetos em *clusters* pequenos

Utilizando Cluster-based Local Outlier Factor (CBLOF) conseguimos identificar *o* e os objetos no cluster C3 como *outliers*

Considera a semelhança entre o objeto e os pontos dos *clusters*



UNIVERSIDADE
PORTUCALENSE

Do conhecimento à prática.