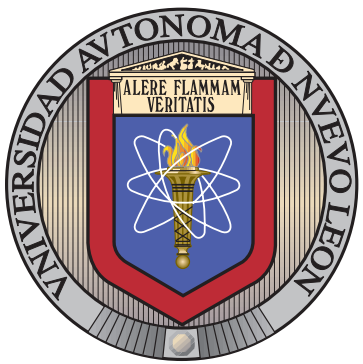
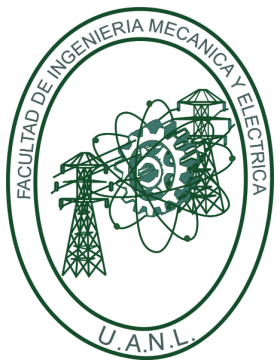
Agrupamiento no supervisado de series de tiempo epidemiológicas de México entre 2005 y 2015

José Alberto Benavides Vázquez

MAESTRÍA EN INGENIERÍA CON ESPECIALIDAD EN INGENIERÍA DE SISTEMAS
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

15 de mayo de 2019





Contenido

Conceptos fundamentales

Semana epidemiológica

- Estándar médico de medición temporal [1]
- Primera semana epidemiológica termina el primer sábado de enero

CIE

- Clasificación Internacional de Enfermedades [44]
- Organización Mundial de la Salud
- Actualmente en la versión 11; periodo 2005-2015, versión 10

Hipótesis

La agrupación a partir de las características de las series de tiempo de los registros semanales de morbilidad en México publicados entre 2005 y 2015 ofrece información estadísticamente significativa que permite **asociar** dichas series de tiempo con una clasificación de referencia con que se etiqueta cada enfermedad.

Objetivos

- Extraer y preparar datos
- Obtener características de las series de tiempo
- Agrupar datos caracterizados
- **Comparar grupos generados con la CIE asignada por la OMS**
- Describir los grupos generados

Metodologías estadísticas

- **Estadística**
  - Descriptiva
    - Medidas de tendencia central
    - Medidas de dispersión
  - **Inferencial**
    - Supervisada
    - **No supervisada**

Algoritmos de agrupamiento para series de tiempo (1979 a 2017)

		Artículos
Métodos	***k*-medias**	[3, 4, 7, 12, 16, 20, 22, 23, 28, 29, 34]
	DTW	[17, 21, 27, 49]
	ARMA	[4, 45, 46]
	ARIMA	[8, 19]
	Jerárquico	[35, 49]
	Red compleja	[13]
Entrada	**ACF**	[3, 11, 12, 43]
	Wavelets	[23, 41, 48]

Origen

1. Descarga iterativa de `PDF`s por patrones en `URL`, p. ej.: `http://www.epidemiologia.salud.gob.mx/doctos/boletin/2008_sem25.pdf`
2. Descarga manual (2013)
3. Descarga de páginas `JPG` comprimidas en `ZIP` (2011)
4. Creación de documento `PDF` con `PDF Mergy` [42] a partir de `JPG`s (2011)

CUADRO 5.1 *Casos por entidad federativa de* ***Enfermedades Infecciosas*** *del Aparato Respiratorio hasta la semana epidemiológica 5; Influenza hasta la 6 del 2012*

ENTIDAD FEDERATIVA	Neumonías y Bronconeumonías CIE-10ª REV. J12-J18 excepto J18.2				Influenza (A H1N1) CIE-10ª REV. J09			Influenza Estacional CIE-10ª REV. J10-J11		
	2012			2011	2012			2012		2011
	Sem.	Acum.		Acum.	Sem.	Acum.		Acum.		Acum.
		M	F			M	F	M	F	
Aguascalientes	79	170	233	677	17	27	19	1	1	5
Baja California	161	445	434	909	18	17	13	6	11	3
Baja California Sur	36	98	91	176	30	74	67	8	8	-
Campeche	20	50	48	86	8	11	16	-	3	-
Coahuila	65	232	217	692	7	10	5	-	-	-
Colima	48	100	111	207	30	50	69	1	1	6
Chiapas	73	160	179	343	21	60	78	9	7	1
Chihuahua	246	654	695	1 632	19	10	18	-	-	6
Distrito Federal	180	806	779	2 154	34	226	282	13	21	65
Durango	69	205	227	625	21	16	14	1	-	-
Guanajuato	184	414	425	1 105	19	22	27	1	1	-
Guerrero	98	210	194	417	9	20	21	2	1	3

Extracción

para cada *directorio en año* **hacer**
- **para cada** *archivo en directorio* **hacer**
  - **si** *termina con* `.pdf` **entonces**
    - leer páginas con `PyPDF2` [31];
    - **para cada** *página en archivo* **hacer**
      - **si** *contiene cuadro de interés* **entonces**
        - extraer contenido con `tabulapy` [2];
        - extraer posiciones del contenido en `JSON`;
        - seleccionar pixeles de columnas de interés;
        - **para cada** *columna en página* **hacer**
          - ajustar anchos de columna;
          - leer filas;
        - **fin**
      - **fin**
    - **fin**
  - **fin**
  - exportar datos en `CSV`;

fin

Resultado

```
2006|02|Tabasco| 'Paratifoidea\r y otras
salmonelosis\rCIE-10a REV.\rA01.1-A02'| '121'
```

784 660 registros

Limpieza

```
for i in range(len(lines)):
  lines[i][1] = lines[i][1][-2:]

  lines[i][4] = str(lines[i][4])
  lines[i][4] = lines[i][4].replace("'", "")
  lines[i][4] = re.sub("^\s+", "", lines[i][4])
  if lines[i][4][0] is "-":
    lines[i][4] = "0"
  if " " in lines[i][4]:
    lines[i][4] = lines[i][4][:lines[i][4].find(" ")]
  lines[i][4] = lines[i][4].replace("n.e", "NA")
  lines[i][4] = lines[i][4].replace("n.d", "NA")
  lines[i][4] = lines[i][4].replace(".", "")
  lines[i][4] = lines[i][4].replace(" ", "")

  lines[i][3] = re.sub("\s+", " ", lines[i][3])
  lines[i][3] = re.sub("^\s+", "", lines[i][3])
  lines[i][3] = lines[i][3].replace("'", "")
  lines[i][3] = lines[i][3].replace("\\r", " ")
  lines[i][3] = str.lower(lines[i][3])
  if "cie" in lines[i][3]:
    cie = lines[i][3][lines[i][3].find("cie-10a rev.") + 13:]
    lines[i][3] = lines[i][3][:lines[i][3].find("cie-10a rev.") - 1]
    lines[i].append(cie)
```

Simplificación y resultado

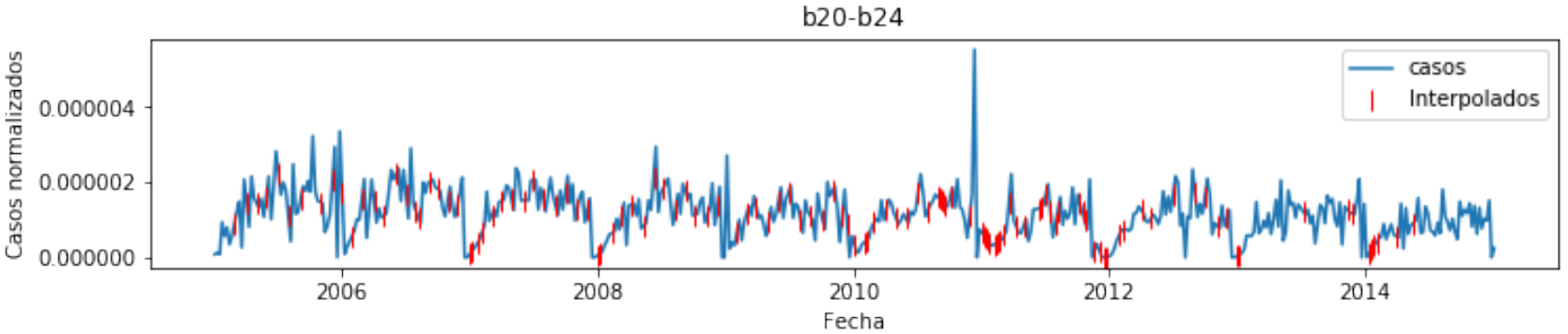
awk

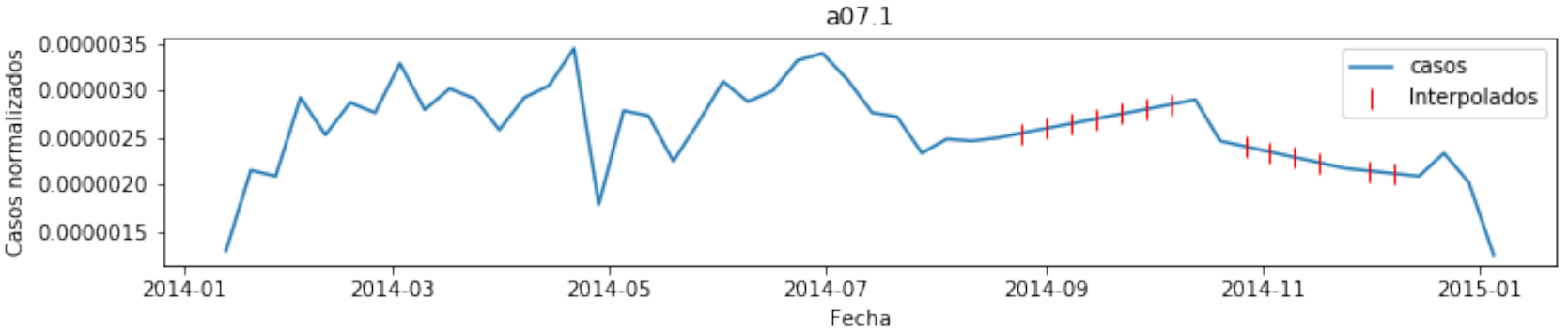
```
awk -F ',' '{print $4, $6 }' sort | uniq -c
```

Resultados nacionales (23 617 registros)

Año	SE	Enfermedad	Casos	CIE
2013	37	Cólera	0	A01
2006	52	Mordeduras por otros mamíferos	117	W55
2014	20	Paludismo por P. Vivax	8	B51

Formato(19 434 registros)



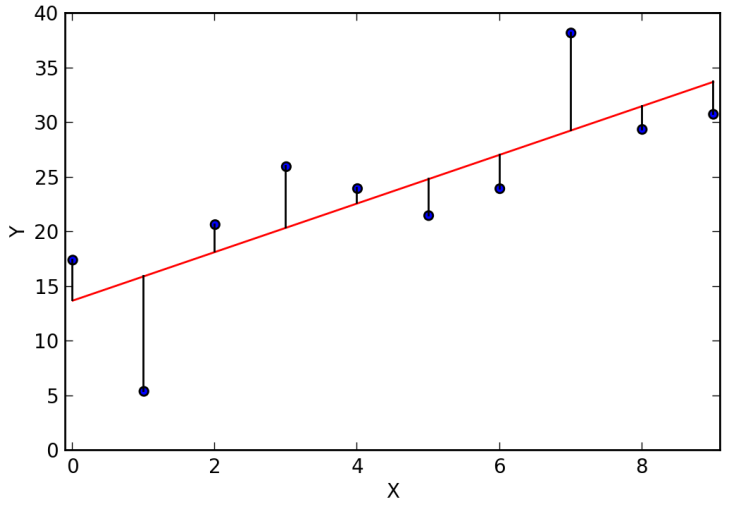


Software para series de tiempo

- `Python v3` [32]: Lenguaje de programación de alto nivel.
- `Pandas` [26]: *Python Data Analysis Library*.
- `datetime` [33]: Manipulación de fechas y horarios.
- `interpolate` [25]: Interpolación por frecuencia semanal.
- `detrend` [39]: Eliminación de la tendencia.

Regresión lineal

Recurso: `linregress` de la librería `SciPy` [40]
$\mathcal{O}(p^2 n + p^3)$ [38]



$$\hat{y}(t, \omega) = \omega_0 + \omega_1 x_1 + \ldots + \omega_t x_t. \tag{1}$$

$$\text{mín} \left(\sum_t (y_t - \hat{y}_t)^2 \right). \tag{2}$$

Tendencia

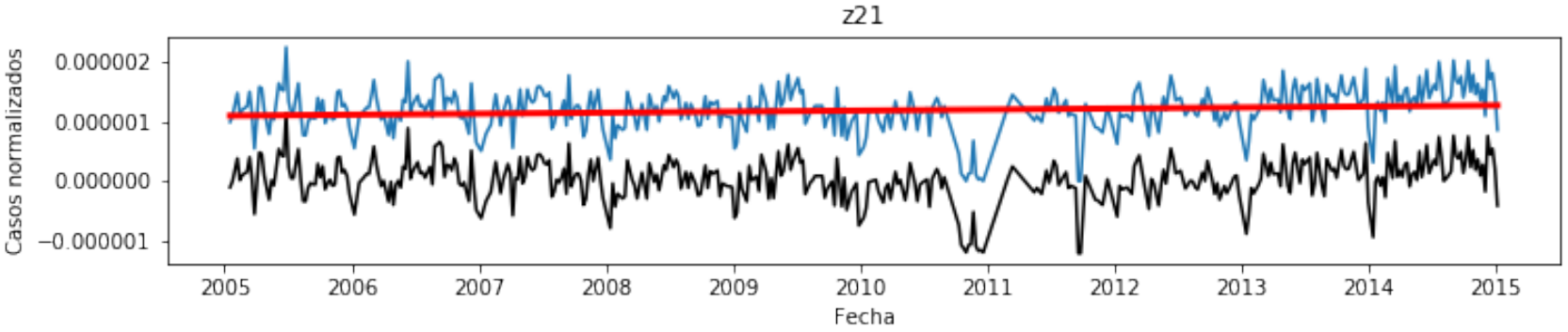


Figura: Infección asintomática por VIH.

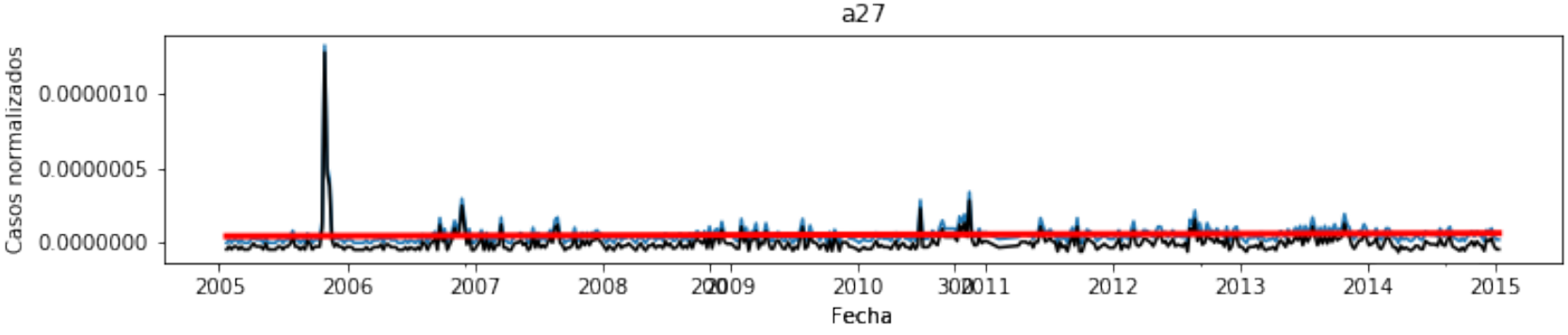


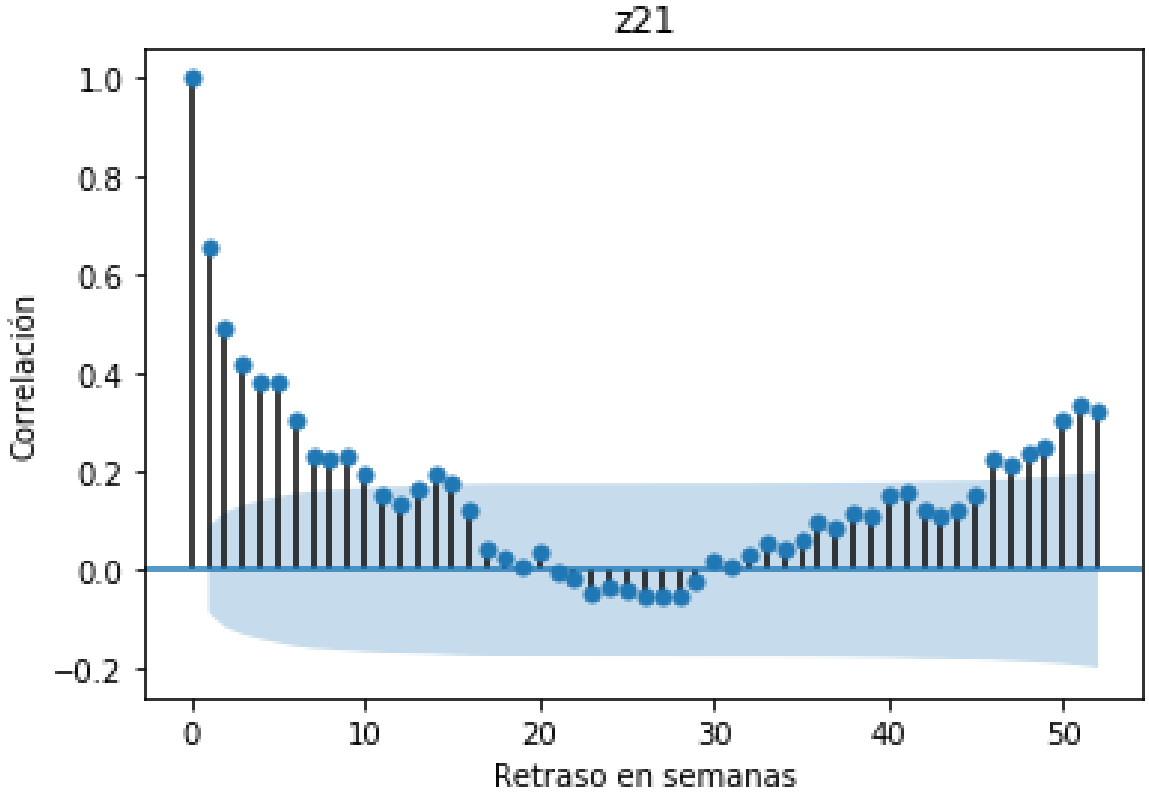
Figura: Tos ferina.

Autocorrelaciones

Recurso: `acf` de la librería `StatsModels` [30]

$$\gamma_x(h) = \text{CoV}(X_{t+h}, X_t) \tag{3}$$

$$\hat{p}_x(h) = \frac{\gamma_x(h)}{\gamma_x(0)} = \text{Cor}(X_{t+h}, X_t). \tag{4}$$



Selección de características

Recurso: *Umbral de varianza* de `scikit-learn` [10];

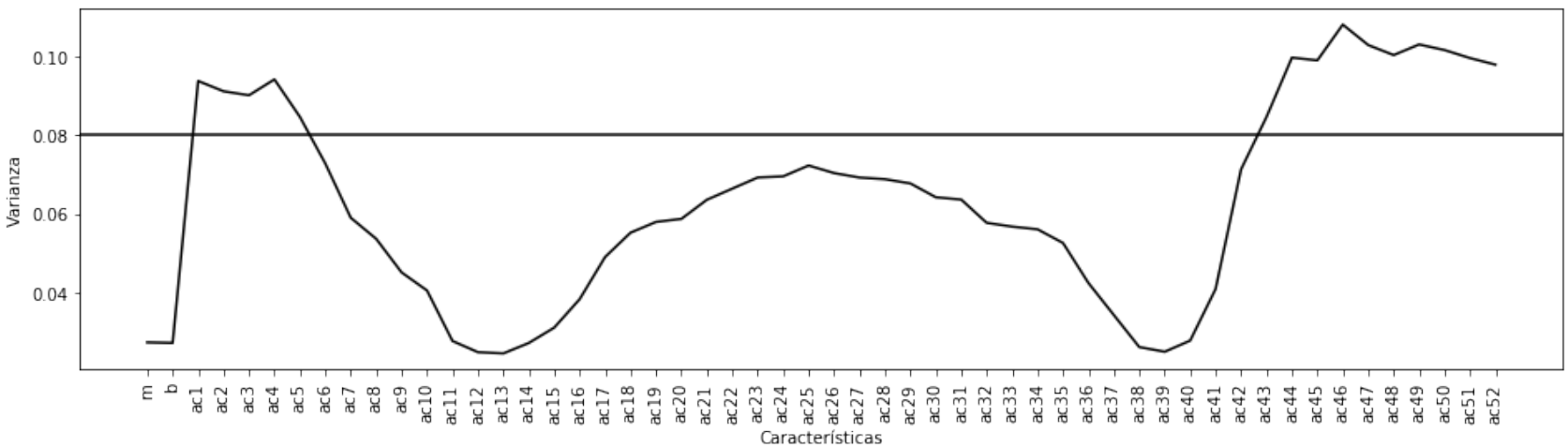


Figura: Características con varianza superior a 0.08

k-medias

Recurso: k-means de scikit-learn [9]

- $X = \{x_i\}, i = 1, \ldots, n$ puntos d-dimensionales
- $C = \{c_j, j = 1, \ldots, k\}$ centros
- μ_j: media de los puntos $x_i \in c_j$

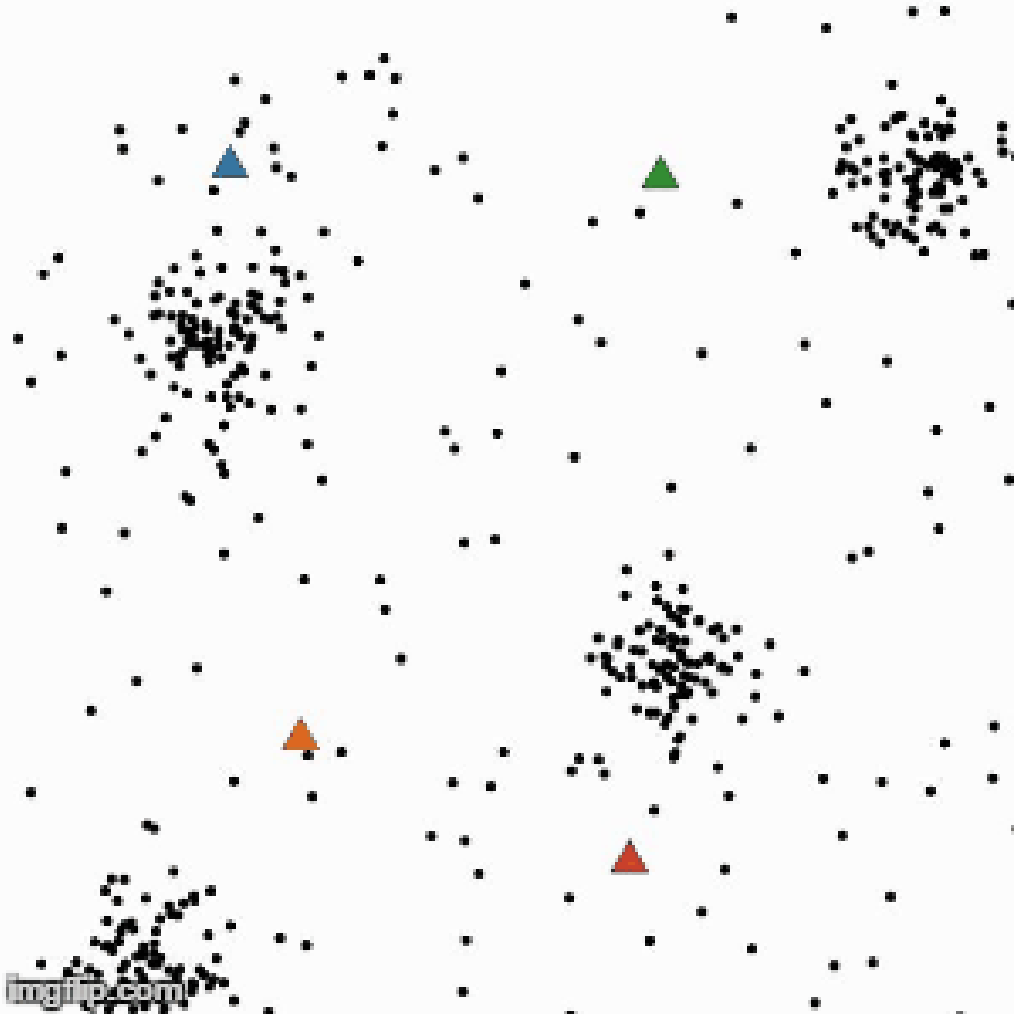
$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} \|x_i - \mu_k\|^2 \tag{5}$$

$$\text{mín}(J(C)) \tag{6}$$

NP-duro [24]

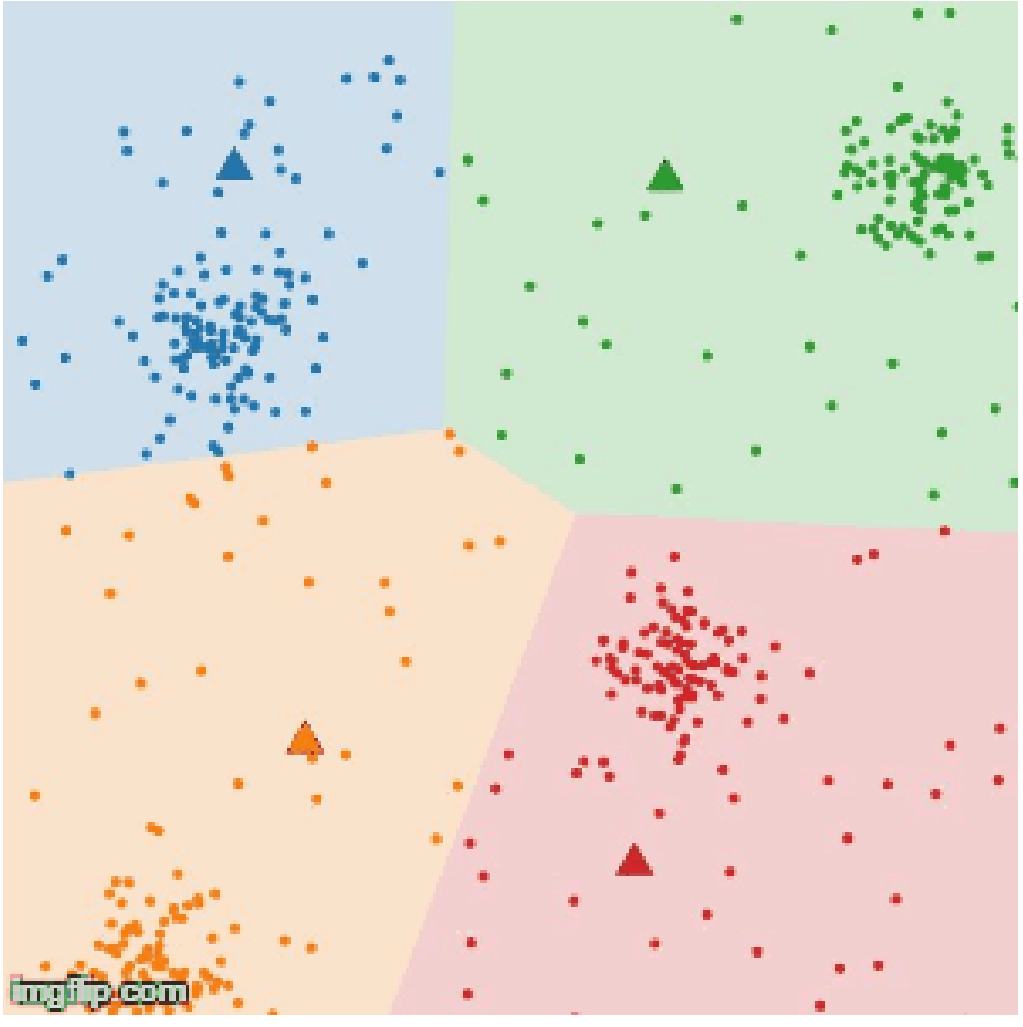
Algoritmo

1. Seleccionar un número k de grupos.
2. Asignarles una posición C_j inicial aleatoria.



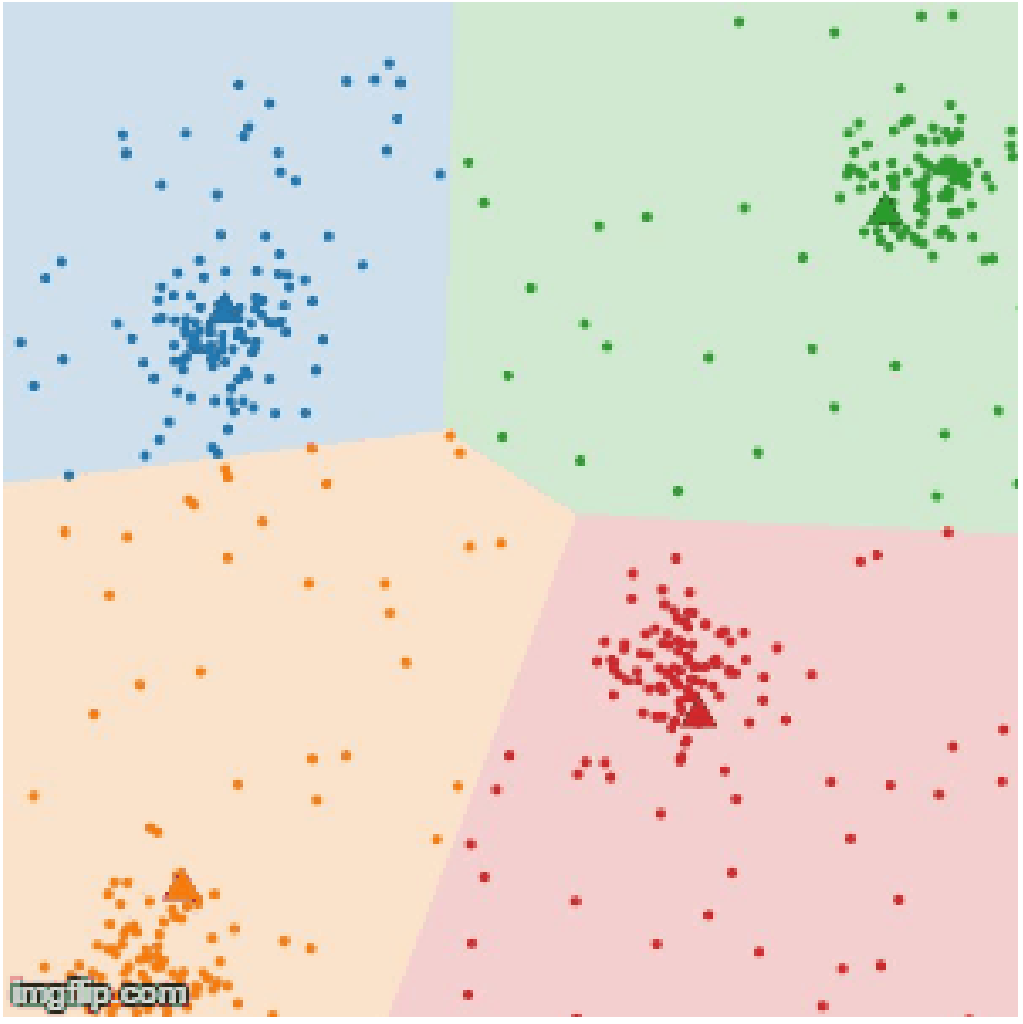
Algoritmo

3. Asociar cada punto con el centro C_j más cercano.



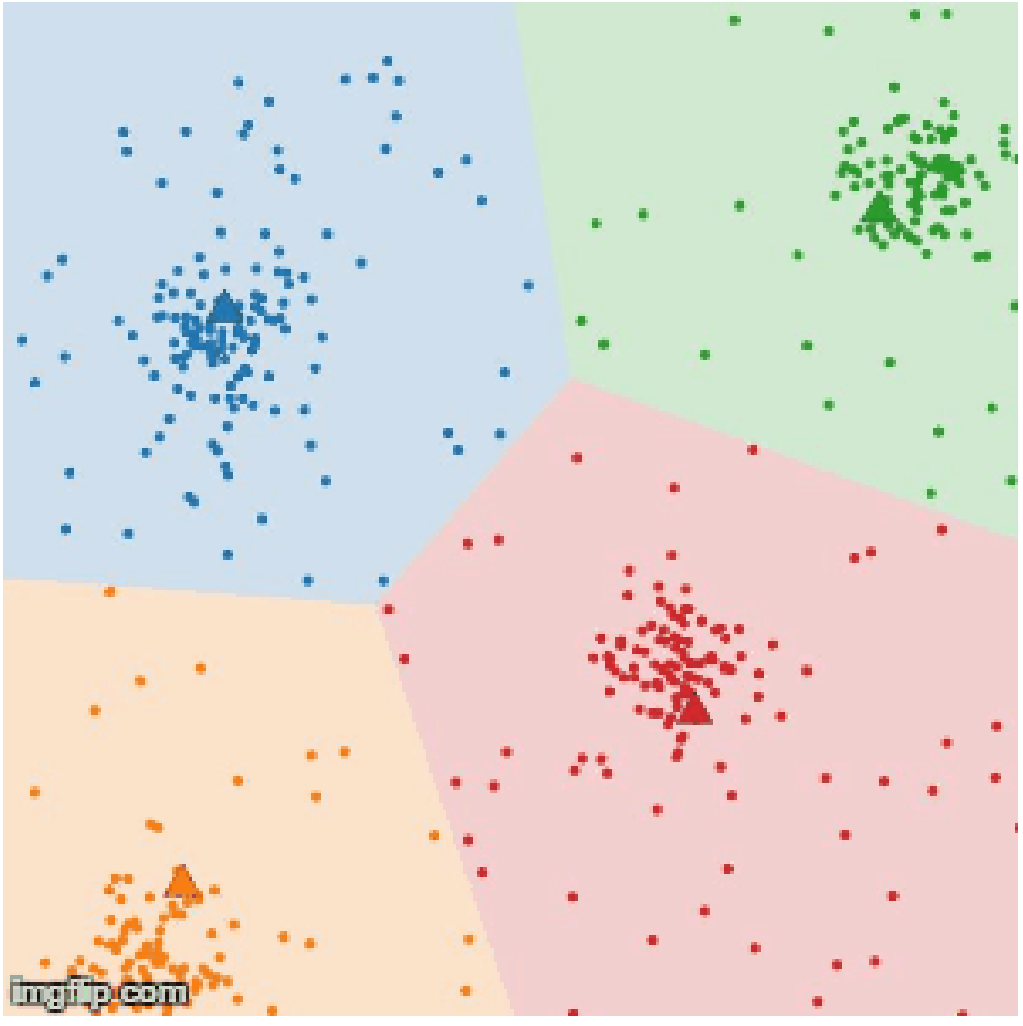
Algoritmo

4. Encontrar la media de cada grupo μ_j.
5. Mover cada centro C_j a dicha media μ_j.



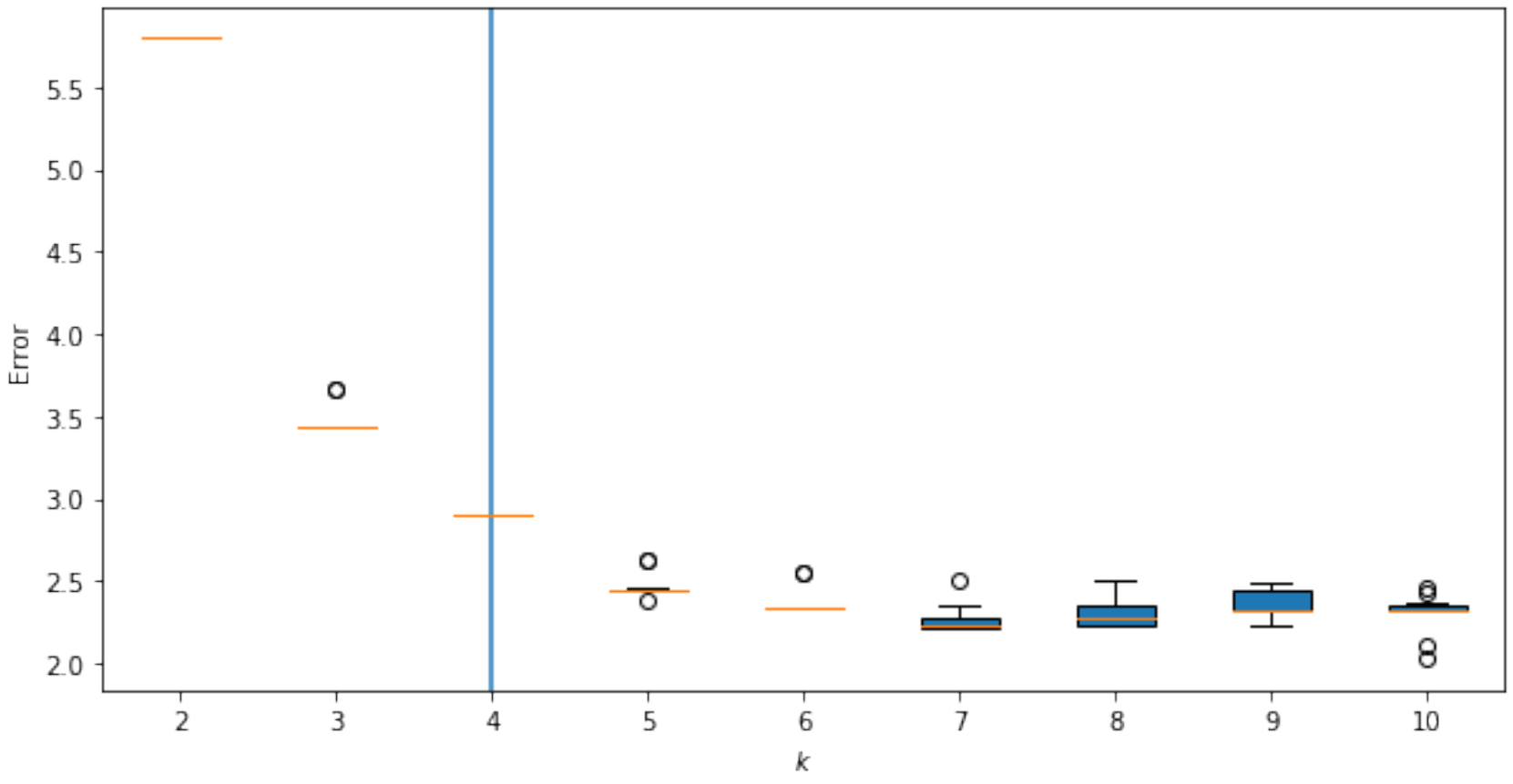
Algoritmo

6. Medir $J(C)$ y si es menor que el anterior, repetir desde el paso 4 [5, 18].



Método del codo [36]

Recurso: `kneedle` [37]



- Comparación con otros algoritmos de agrupamiento.
- Usar otras características y medidas de distancias.
- Comparar con registros diarios de consultas.
- Usar grupos resultantes para mejorar algoritmos de clasificación.

Gracias por su atención.

jose.benavidesvz@uanl.edu.mx

[1] Arias, J. R. (2006). What is an epidemiological week and why do we use them? *The Seeker*, 6(1):7.

[2] Ariga, A. (2018). chezou/tabula-py: Simple wrapper of tabula-java: extract table from pdf into pandas dataframe. Accedido: 2018-07-01.

[3] Ashish Singhal, D. E. (2002). Clustering of multivariate time-series data. In *Proceedings of the 2002 American Control Conference*, pages 273–280, Arkansas. IEEE.

[4] Bagnall, A. and Janacek, G. (2005). Clustering time series with clipped data. *Machine Learning*, 58(2):151–178.

[5] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, Singapore.

[6] Brockwell, P. J. and Davis, R. A. (2016). *Introduction to Time Series and Forecasting*. Springer, Switzerland.

[7] Chen, J. R. (2005). Making subsequence time series clustering meaningful. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*.

[8] Corduas, M. and Piccolo, D. (2008). Time series clustering and classification by the autoregressive metric. *Computational Statistics and Data Analysis*, 52(4):1860 – 1872.

[9] Desarrolladores de scikit-learn (2019a). 2.3.2. *k*-means. Accedido: 2019-03-12.

[10] Desarrolladores de scikit-learn (2019b). sklearn.feature_selection.variancethreshold. Accedido: 2019-03-22.

[11] D'Urso, P. and Maharaj, E. A. (2009). Autocorrelation-based fuzzy clustering of time series. *Fuzzy Sets and Systems*, 160(24):3565 – 3589. Theme: Non-Linear Systems and Fuzzy Clustering.

[12] Ernst, J., J. Nau, G., and Bar-Joseph, Z. (2005). Clustering short time series gene expression data. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 285–289, New York. ACM.

[13] Ferreira, L. N. and Zhao, L. (2015). Time Series Clustering via Community Detection in Networks. *arXiv e-prints*.

[14] Free Software Foundation (2011). Gawk - gnu project - free software foundation (fsf). Accedido: 02-02-2019.

[15] Fulcher, B. D. and Jones, N. S. (2014). Highly comparative feature-based time-series classification. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):3026–3037.

[16] Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A *k*-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

[17] Izakian, H., Pedrycz, W., and Jamal, I. (2015). Fuzzy clustering of time series data using dynamic time warping distance. *Engineering Applications of Artificial Intelligence*, 39:235 – 244.

[18] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 13(8):651–666.

[19] Kalpalis, K., Gada, D., and Puttagunta, V. (2001). Distance measures for effective clustering of arima time-series. In *Proceedings 2001 IEEE International Conference on Data Mining*, pages 273–280, California. IEEE.

[20] Keogh, E. and Lin, J. (2005). Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems*, 8(2):154–177.

[21] Keogh, E. J. and Pazzani, M. J. (2000). Scaling up dynamic time warping for datamining applications. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 285–289, New York. ACM.

[22] Lai, R. K., Fan, C.-Y., Huang, W.-H., and Chang, P.-C. (2009). Evolving and clustering fuzzy decision tree for financial time series data forecasting. *Expert Systems with Applications*, 36(2, Part 2):3761 – 3773.

[23] Lin, J., Vlachos, M., Keogh, E., and Gunopulos, D. (2004). Iterative incremental clustering of time series. In *Advances in Database Technology - EDBT 2004*, pages 106–122, Berlin. Springer.

[24] Mahajan, M., Nimbhorkar, P., and Varadarajan, K. (2012). The planar *k*-means problem is np-hard. *Theoretical Computer Science*, 442:13 – 21. Special Issue on the Workshop on Algorithms and Computation (WALCOM 2009).

[25] NumFOCUS (2019a). pandas.series.interpolate. Accedido: 2019-03-22.

[26] NumFOCUS (2019b). Python data analysis library. Accedido: 2019-04-07.

[27] Oates, T. (1999). Identifying distinctive subsequences in multivariate time series by clustering. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, pages 322–326, New York, NY, USA. ACM.

[28] Paparrizos, J. and Gravano, L. (2015). k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, SIGMOD '15, pages 1855–1870, New York, NY, USA. ACM.

[29] Paparrizos, J. and Gravano, L. (2017). Fast and accurate time-series clustering. *ACM Trans. Database Syst.*, 42(2):8:1–8:49.

[30] Perktold, J., Seabold, S., and Taylor, J. (2019). statsmodels.tsa.stattools.acf. Accedido: 2019-04-07.

[31] Phaseit Inc. and Mathieu Fenniak (2016). PyPDF2 Documentation. Accedido: 02-07-2018.

[32] Phython Software Foundation (2018). Python 3.7.0. Accedido: 2018-08-13.

[33] Python Software Foundation (2019). datetime – basic date and time types. Accedido: 2019-04-07.

[34] Ratanamahatana, C., Keogh, E., Bagnall, A. J., and Lonardi, S. (2005). A novel bit level time series representation with implication of similarity search and clustering. In *Advances in Knowledge Discovery and Data Mining*, pages 771–777, Berlin. Springer Berlin Heidelberg.

[35] Rodrigues, P. P., Gama, J., and Pedroso, J. P. (2008). Hierarchical clustering of time-series data streams. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):615–627.

[36] Salvador, S. and Chan, P. (2004). Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *16th IEEE International Conference on Tools with Artificial Intelligence*, pages 576–584.

[37] Satopaa, V., Albrecht, J., Irwin, D., and Raghavan, B. (2011). Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171.

[38] The Kernel Trip (2019). Computational complexity of machine learning algorithms. Accedido: 2019-05-11.

[39] The SciPy community (2019a). scipy.signal.detrend. Accedido: 2019-04-07.

[40] The SciPy community (2019b). scipy.stats.linregress – scipy v1.2.1 reference guide. Accedido: 04-07-2019.

[41] Vlachos, M., Lin, J., Keogh, E., and Gunopulos, D. (2003). A wavelet-based anytime algorithm for *k*-means clustering of time series. *Proc. Workshop on Clustering High Dimensionality Data and its Applications*.

[42] w69b (2018). PDF Mergy - WebApp to merge PDF files. Accedido: 2018-11-23.

[43] Wang, X., Smith, K., and Hyndman, R. (2006). Characteristic-based clustering for time series data. *Data Mining and Knowledge Discovery*, 13(3):335–364.

[44] World Health Organization (2018). WHO — international classification of diseases, 11th revision (icd-11). Accedido: 2018-09-30.

[45] Xiong, Y. and Yeung, D.-Y. (2002). Mixtures of arma models for model-based time series clustering. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings*, pages 717–720, Maebashi. IEEE.

[46] Xiong, Y. and Yeung, D.-Y. (2004). Time series clustering with arma mixtures. *Pattern Recognition*, 37(8):1675 – 1689.

[47] Yildiz, B., Kaiser, K., and Miksch, S. (2005). pdf2table: A method to extract table information from pdf files. In *IICAI*.

[48] Zhang, H., Ho, T., Zhang, Y., and Lin, S. (2006). Unsupervised feature extraction for time series clustering using orthogonal wavelet transform. *Informatica (Slovenia)*, 30:305–319.

[49] Zhang, X., Liu, J., Du, Y., and Lv, T. (2011). A novel clustering method on time series data. *Expert Systems with Applications*, 38(9):11891 – 11900.