

PCA Robusto, Outliers y Distancia de Mahalanobis

El Análisis de Componentes Principales (PCA) es una técnica de reducción de dimensionalidad que permite transformar un conjunto de variables posiblemente correlacionadas en un nuevo sistema ortogonal de ejes, llamados componentes principales, que capturan la mayor parte de la variabilidad presente en los datos. Sin embargo, su aplicación estándar que se basa en la matriz de covarianza empírica es altamente sensible a la presencia de outliers. Esto hace que incluso una sola observación atípica puede distorsionar notablemente la orientación de los componentes, afectando tanto la interpretación como el desempeño de cualquier modelo que se entrene sobre estos nuevos ejes. Por ello, surge la necesidad de un enfoque más robusto frente a estas anomalías.

Detección de outliers y la necesidad de una medida multivariada

Antes de aplicar un PCA robusto, es necesario identificar y caracterizar las observaciones atípicas dentro del conjunto de datos. Para ello, se requiere una métrica de distancia que sea apropiada en un contexto multivariado, es decir, que considere la distribución global de los datos, tome en cuenta la correlación entre las variables y sea invariante a la escala de las variables.

Como primera opción y de manera intuitiva, es posible pensar en usar la distancia euclídea; sin embargo, no satisface ninguna de estas condiciones de forma adecuada. En efecto, observe que:

1. **No considera la forma de la distribución:** La distancia euclídea simplemente mide la “recta” entre dos puntos en el espacio, y esto no tiene en cuenta si los datos están más dispersos en una dirección que en otra. Lo anterior puede llevar a subestimar o sobreestimar distancias en presencia de variables con alta dispersión o estructuras elongadas.
2. **Ignora la correlación entre variables:** Suponga que dos variables están altamente correlacionadas. En este caso, una combinación de ambas no aporta mucha información nueva, pero la distancia euclídea las trata como si fueran independientes, duplicando artificialmente su peso. Esto puede llevar a errores en la detección de outliers, ya que los puntos que siguen esa correlación natural podrían parecer más “lejanos” de lo que realmente son.
3. **Es sensible a la escala de las variables:** Si una variable tiene valores que van de 0 a 1 y otra de 0 a 10000, la segunda dominará completamente el cálculo de distancia. A menos que se realice un escalado previo, la distancia euclídea no permite una comparación justa entre dimensiones. Incluso con escalado, sigue sin capturar correlaciones entre atributos.

Por esta razón, se introduce la distancia de Mahalanobis, una alternativa más adecuada, ya que ajusta el espacio de representación según la estructura estadística de los datos. Grosso modo, usa la matriz de covarianza para “transformar” el espacio, rotando y escalando los ejes de forma que todas las variables sean comparables y sus correlaciones sean tenidas en cuenta.

La distancia de Mahalanobis

La distancia de Mahalanobis, propuesta por el estadístico indio P. C. Mahalanobis en 1936, cuantifica qué tan lejos se encuentra un punto respecto al “centro” de una distribución multivariada, considerando su forma y dispersión.

Formalmente, para un vector de características X , el vector μ de medias (una media por cada variable) que representa el centroide de la distribución multivariada y la matriz de covarianza se define como

$$D_M(X) = \sqrt{(X - \mu)^T \Sigma^{-1} (X - \mu)}$$

donde

- $X - \mu$ es un vector de desplazamiento respecto al centro
- Σ^{-1} permite escalar y rotar el espacio para que la dispersión en todas las direcciones quede “igualada”

El producto escalar resultante da una medida de “lejanía” normalizada según la geometría de la nube de datos.

Finalmente, esta distancia es clave para detectar outliers: aquellos puntos con una distancia de Mahalanobis alta pueden ser considerados como anómalos en el contexto de todas las variables simultáneamente.

Ejemplo ilustrativo (basado en Wikipedia)

Para comprender intuitivamente la utilidad de la distancia de Mahalanobis frente a la distancia euclídea, se recomienda consultar el ejemplo clásico descrito en la entrada de [Wikipedia sobre la distancia de Mahalanobis](#).

En dicho ejemplo, se plantea el caso de un pescador que desea clasificar salmones según sus características físicas: longitud y anchura. Como estas variables tienen diferentes escalas y varianzas, aplicar directamente la distancia euclídea conduce a una clasificación sesgada, ya que otorga más peso a la variable con mayor variabilidad (la longitud). La distancia de Mahalanobis, al tener en cuenta la varianza y la correlación entre variables, permite realizar una comparación más justa y representativa entre observaciones, ponderando adecuadamente las diferencias en cada dimensión.

Consideraciones sobre la invertibilidad de la Matriz de Covarianza y su Estimación Robusta

El análisis de componentes principales (PCA) y la distancia de Mahalanobis requieren, en su formulación matemática, la inversión de la matriz de covarianza del conjunto de datos. No obstante, la inversa de la matriz de covarianza no siempre existe, y este detalle técnico tiene profundas implicaciones prácticas en el tratamiento de datos reales.

Comencemos recordando que la matriz de covarianza es invertible si y sólo si es no singular, lo que significa que su determinante es distinto de cero o, equivalentemente, que tiene rango completo, es decir, todas sus columnas (o filas) son linealmente independientes.

Esto no se cumple en los casos donde hay variables correlacionadas o redundantes, pues si una o más variables son combinaciones lineales exactas de otras, la matriz pierde rango (por ejemplo, si una columna es el doble de otra) o donde hay pocas muestras respecto al número de variables, ya que si el número de observaciones n es menor (o apenas comparable) al número de variables p , la matriz de covarianza será de rango deficiente. Este problema es común en contextos de alta dimensionalidad como genómica o visión por computador.

A continuación, se describen algunas estrategias desarrolladas en la literatura para abordar este problema, agrupadas según los supuestos asumidos y las herramientas matemáticas utilizadas:

1) Regularización por Ridge o Shrinkage Estimators

Una estrategia directa consiste en regularizar la matriz de covarianza muestral sumando una constante positiva λ a su diagonal:

$$\Sigma_{reg} = \Sigma + \lambda I$$

Esto garantiza que la matriz resultante sea positiva definida y, por tanto, invertible, aún si la matriz original no lo era. Esta idea subyace en métodos ampliamente utilizados como ridge regression o los estimadores de shrinkage, y ha sido fundamental en aplicaciones como la optimización de portafolios financieros (Deng y Tsui, 2013).

2) Estimación Parcial mediante Decomposición de Cholesky Modificada (MCD)

En contextos como datos longitudinales, series temporales o señales espectrales, donde las variables tienen un orden natural, se puede explotar esta estructura mediante una decomposición de Cholesky modificada, como la propuesta por

Pourahmadi (1999). Este enfoque garantiza positividad definida y proporciona una interpretación en términos de regresión secuencial.

En particular, Kang, Lian y Deng (2021) proponen un modelo de Block Cholesky Decomposition (BCD) que extiende este enfoque cuando se conoce parcialmente el orden de las variables. El BCD permite mantener la interpretabilidad y garantizar la validez estadística del estimador, incorporando además técnicas de regularización para fomentar la sparsity (es decir, matrices con muchos ceros), lo cual es fundamental para aplicaciones en modelos gráficos y selección de variables.

3) Selección de Variables y Reducción Dimensional

Otra estrategia clásica es eliminar variables redundantes o altamente correlacionadas antes de estimar la covarianza. Esto mejora el *condicionamiento* de la matriz y evita problemas de singularidad. Sin embargo, este enfoque puede resultar riesgoso si se eliminan variables informativas. En el contexto del presente proyecto, esta estrategia no fue utilizada, ya que se busca mantener la estructura completa de los datos y abordar directamente el problema de los outliers mediante estimación robusta.

4) Estimación Robusta: Minimum Covariance Determinant (MCD)

Cuando el problema no es la colinealidad per se, sino la presencia de outliers que distorsionan las estimaciones, la solución más adecuada es utilizar un estimador robusto como el Minimum Covariance Determinant (MCD), propuesto por Rousseeuw y Van Driessen (1999). El MCD busca el subconjunto de tamaño h que tenga la menor determinante de la covarianza, excluyendo los valores extremos.

El algoritmo FAST-MCD implementado en *scikit-learn* permite su aplicación eficiente incluso en alta dimensionalidad. La matriz estimada mediante MCD es típicamente invertible, ya que se calcula sobre un subconjunto más "estable" de observaciones, excluyendo aquellas que causan problemas de colinealidad o singularidad.

MCD no sólo permite calcular la inversa de la matriz de covarianza en presencia de outliers, sino que también proporciona la media robusta y las distancias de Mahalanobis robustas, las cuales son fundamentales para la detección de atípicos y el desarrollo de PCA robusto.

5) Modelos Basados en Optimización Convexa (Lasso y Programación Lineal)

En contextos de alta dimensión, algunos métodos recientes formulan la estimación de la inversa de la matriz de covarianza como un problema de optimización convexa. Por ejemplo, Yuan (2007) explora la conexión entre la regresión multivariada y la matriz de precisión, proponiendo un método basado en programación lineal que explota la sparsity estructural en la matriz inversa. Este enfoque es especialmente

útil en modelos gráficos gaussianos, donde la estructura de ceros en la inversa de la covarianza corresponde a relaciones de independencia condicional entre variables.

Estimación Robusta con Minimum Covariance Determinant (MCD)

Entre las estrategias anteriormente mencionadas se hizo uso de la del estimador Minimum Covariance Determinant (MCD) y es que además de reducir significativamente la influencia de los outliers, proporcionar una matriz de covarianza invertible y bien condicionada, al evitar puntos que generan colinealidades artificiales y preservar la estructura real del conjunto de datos, facilitando técnicas de análisis posteriores como el PCA robusto y la distancia de Mahalanobis robusta, el MCD ofrece una serie de ventajas con respecto a otros métodos robustos que se detallarán más adelante.

Funcionamiento del MCD

El MCD se puede describir conceptualmente en los siguientes pasos:

1. Generación de subconjuntos candidatos: A partir del conjunto total de datos X , el algoritmo genera múltiples subconjuntos de tamaño h , donde h es típicamente el 75% del número total de muestras. Este tamaño es lo suficientemente grande para captar la estructura subyacente de los datos, pero lo bastante pequeño para excluir puntos extremos.
2. Evaluación de subconjuntos: Para cada subconjunto, se calcula la media muestral, la matriz de covarianza y el determinante de dicha matriz. El determinante de la matriz de covarianza actúa como una medida del volumen de dispersión de ese subconjunto: cuanto menor sea, más compacto y coherente se considera el grupo de observaciones.
3. Selección del subconjunto óptimo: Se selecciona el subconjunto cuyo determinante de la covarianza es el mínimo entre todos los candidatos. Este subconjunto es interpretado como el núcleo más representativo de los datos.
4. Estimación robusta: Finalmente, a partir de ese subconjunto óptimo, se calcula una media robusta que representa el centro de la distribución sin influencia de outliers y una matriz de covarianza robusta que refleja la dispersión real de los datos centrales.

Comparación con Otros Estimadores Robustos

El MCD pertenece a una familia de métodos robustos de estimación de covarianza. A continuación, se comparan algunas de las principales alternativas:

- **Minimum Volume Ellipsoid (MVE):** Este método busca el elipsoide de menor volumen que contenga una fracción del conjunto de datos. Aunque conceptualmente similar al MCD, MVE es más sensible numéricamente y más costoso computacionalmente (Maronna et al., 2006).
- **S-estimadores:** Basados en minimizar funciones de pérdida robustas, como la función bisquare de Tukey, estos métodos permiten una gran flexibilidad. Sin embargo, su rendimiento depende críticamente de la elección del parámetro de tuning que controla la sensibilidad al outlier. Esto introduce incertidumbre en el análisis si no se ajustan adecuadamente (Maronna et al., 2006).
- **Shrinkage estimators:** Estos métodos combinan la matriz empírica con una matriz objetivo (por ejemplo, la identidad) mediante una interpolación. Aunque útiles para evitar problemas de singularidad, no son inherentemente robustos a outliers, ya que aún utilizan toda la muestra sin excluir valores atípicos (Ledoit & Wolf, 2004).

Pese a la variedad de opciones, el MCD representa una opción sólida para este proyecto por las siguientes razones:

- Tiene un breakdown point, o proporción máxima de valores atípicos que el estimador puede tolerar antes de dar resultados arbitrarios, alto (hasta el 50%), lo que le confiere alta robustez ante outliers (Rousseeuw, 1984).
- Proporciona una estimación directa de la media y la covarianza robustas.
- Es compatible de manera natural con PCA robusto, ya que permite aplicar el análisis solo sobre el subconjunto central de datos (Hubert et al., 2005).
- Su implementación en scikit-learn es estable, reproducible y computacionalmente eficiente gracias al algoritmo FAST-MCD.
- No requiere una elección delicada de parámetros como los S-estimadores, lo cual facilita su integración sin sobreajuste.

Si bien existen estimadores alternativos con ventajas específicas en ciertos contextos (como menor varianza asintótica o mayor flexibilidad), el MCD ofrece un balance ideal entre robustez, simplicidad y aplicabilidad práctica, siendo especialmente adecuado para análisis multivariados en presencia de ruido y outliers como el de este proyecto.

En comparación, MCD ofrece un excelente balance entre robustez, interpretabilidad y aplicabilidad multivariada, por lo cual ha sido ampliamente adoptado como técnica estándar en detección de outliers y análisis exploratorio robusto.

Sin embargo, pese a sus múltiples ventajas, el MCD presenta ciertas limitaciones:

- **Costo computacional:** El cálculo del MCD puede resultar exigente en términos de tiempo y memoria cuando se aplica a conjuntos de datos con gran número de observaciones o variables. Esto se debe a que implica buscar múltiples subconjuntos posibles y calcular determinantes de matrices de covarianza. Sin embargo, en este proyecto, el tamaño del dataset es **moderado** y se encuentra dentro de los rangos para los que el algoritmo **FAST-MCD** (véase la siguiente sección) de *scikit-learn* se desempeña eficientemente (Rousseeuw & Van Driessen, 1999).
- **Elección del parámetro hhh:** Este parámetro define cuántas observaciones se consideran para calcular la matriz de covarianza robusta (típicamente entre el 70% y 90% del total). Si se elige un h muy pequeño, se gana robustez pero se pierde eficiencia; si se elige muy grande, se incluyen más outliers. En este caso, se ha utilizado el valor por defecto recomendado por *scikit-learn*, que optimiza automáticamente el valor de h para balancear robustez y eficiencia según el tamaño de la muestra y la dimensión de los datos.
- **Sensibilidad a múltiples clusters o densidades:** En datasets con estructuras internas complejas, como varios grupos con distintas distribuciones o densidades, el MCD podría seleccionar un subconjunto representativo de solo uno de ellos, ignorando los demás. Sin embargo, esto no es un problema en este proyecto, ya que el conjunto de datos no presenta alta heterogeneidad en cuanto a estructuras internas o múltiples distribuciones separadas, el problema de clasificación abordado tiene una variable objetivo (target) balanceada, lo que garantiza que cada clase está representada adecuadamente en el conjunto de entrenamiento y prueba y además, la partición de los datos se realiza con `train_test_split(..., stratify=y)`, lo que preserva la proporción de clases en ambos conjuntos, reduciendo el riesgo de que una clase quede subrepresentada al aplicar la detección de outliers o el PCA robusto sobre el subconjunto "limpio".

Implementación eficiente: FAST-MCD

Aunque el enfoque descrito en la sección sobre el funcionamiento del MCD es conceptualmente directo, la búsqueda exhaustiva de todos los subconjuntos posibles de tamaño h sería computacionalmente inviable, especialmente para grandes conjuntos de datos. Para abordar este problema, Rousseeuw y Van

Driessen (1999) propusieron el algoritmo FAST-MCD, una variante eficiente que reduce enormemente la carga computacional manteniendo una alta calidad de estimación.

FAST-MCD no evalúa todos los subconjuntos posibles, sino que utiliza un procedimiento iterativo y aleatorizado basado en inicializaciones inteligentes, reestimaciones y pasos de afinamiento (refinement steps) que convergen rápidamente hacia un subconjunto cercano al óptimo global. Gracias a esta estrategia, FAST-MCD puede escalar a conjuntos de datos de dimensión media (cientos de muestras y decenas de variables), y es el enfoque utilizado en scikit-learn mediante la clase `sklearn.covariance.MinCovDet`.

PCA Robusto

El análisis clásico de componentes principales (PCA) es una técnica estadística fundamental para la reducción de dimensionalidad, ampliamente utilizada por su capacidad para extraer las direcciones de mayor varianza en los datos. Sin embargo, como se ha venido hablando, su aplicación directa presenta una debilidad crítica: la sensibilidad extrema a valores atípicos (outliers). Esto se debe a que el PCA se basa en la estimación de la matriz de covarianza empírica, la cual puede ser distorsionada significativamente por un pequeño número de observaciones anómalas, lo que afecta tanto la orientación de las componentes principales como la calidad de la compresión y proyección obtenida.

Para subsanar esta limitación y preservar la estructura inherente del conjunto de datos, se adoptó un enfoque robusto que integra la detección de outliers basada en la distancia de Mahalanobis con un procedimiento de escalado robusto y una aplicación posterior de PCA. Este flujo de trabajo combina técnicas estadísticas robustas con métodos de reducción de dimensionalidad, mejorando la estabilidad de las proyecciones en presencia de ruido extremo.

En primer lugar, se aplica el estimador de covarianza robusta Minimum Covariance Determinant (MCD), implementado mediante la clase `MinCovDet` de scikit-learn, que calcula una media y una matriz de covarianza resistentes a la influencia de outliers. Este estimador opera restringiéndose a un subconjunto de las observaciones (típicamente el 75% menos atípico), buscando minimizar el determinante de la covarianza dentro de ese subconjunto, lo cual maximiza la concentración de datos y excluye puntos aberrantes.

Con esta covarianza robusta $\Sigma_{robusta}$ y media robusta $\mu_{robusta}$, se calcula la distancia de Mahalanobis para cada observación x_i

$$D_M(x_i) = \sqrt{(x_i - \mu_{robusta})^T \Sigma_{robusta}^{-1} (x_i - \mu_{robusta})}$$

Esta medida multivariante permite cuantificar qué tan alejada está una observación de la distribución central robusta. Se define un umbral de corte mediante el percentil 97.5% de la distribución χ^2 con k grados de libertad (donde k es la cantidad de variables numéricas). Aquellas observaciones con distancias por encima de este umbral son marcadas como outliers multivariantes.

Una vez identificados los outliers, se procede a su eliminación para obtener un subconjunto “limpio” del conjunto de entrenamiento, con el objetivo de evitar que valores atípicos influyan en los pasos posteriores. Luego, se realiza un escalado robusto de las variables numéricas, centrando cada dimensión en su media robusta y normalizando por la desviación estándar derivada de la matriz de covarianza robusta (i.e., usando la raíz cuadrada de la diagonal de $\Sigma_{robusta}$):

$$X_{esc} = \frac{X - \mu_{robusta}}{\sqrt{diag(\Sigma_{robusta})}}$$

Este paso garantiza que las variables numéricas estén centradas y comparables en magnitud, sin depender de estadísticas sensibles a outliers como la media y desviación estándar empíricas.

Finalmente, se aplica el PCA clásico sobre los datos robustamente escalados, utilizando el subconjunto libre de outliers como conjunto de entrenamiento. A pesar de emplear el algoritmo estándar de PCA (basado en descomposición en valores singulares o SVD), el hecho de que los datos hayan sido previamente filtrados y transformados mediante estimaciones robustas permite que las componentes principales capturen las direcciones de mayor variación estructural del conjunto, sin estar sesgadas por ruido extremo.

Este enfoque puede considerarse una forma híbrida de PCA robusto, donde la robustez no proviene de modificar internamente el algoritmo de PCA, sino de una fase de preprocesamiento estadísticamente robusta, que asegura que los supuestos del PCA clásico sean razonablemente satisfechos en la práctica.

Experimentación

Cabe señalar que en etapas previas del proyecto no fue posible aplicar de manera extensiva las técnicas de reducción de dimensionalidad sobre el conjunto de datos completo, debido a limitaciones computacionales en el entorno local de trabajo. Estas restricciones obligaron a realizar tanto el análisis exploratorio como los entrenamientos iniciales sobre subconjuntos reducidos del dataset, lo cual, si bien permitió establecer una línea base funcional, comprometía en cierta medida la generalización y profundidad del análisis.

Sin embargo, en esta fase final del estudio se logró disponer de una infraestructura más robusta mediante el uso de Google Colab Pro, lo que permitió aprovechar recursos computacionales adicionales como aceleración por GPU y mayor capacidad de memoria RAM. Esta mejora hizo posible la ejecución del pipeline completo de preprocesamiento y entrenamiento sobre la totalidad del conjunto de datos, incluyendo técnicas demandantes como la estimación robusta de covarianza mediante el algoritmo FAST-MCD, la detección de outliers multivariantes basada en distancias de Mahalanobis, y la posterior reducción de dimensionalidad con PCA sobre datos robustamente transformados.

Este salto en capacidad computacional no solo habilitó un análisis más exhaustivo y realista, sino que también permitió evaluar el impacto de estos métodos robustos en la calidad predictiva de distintos modelos de clasificación, ofreciendo conclusiones más sólidas y representativas.

Análisis exploratorio y diseño del preprocesamiento

Dentro de la fase experimental, como primer paso se desarrolló un análisis exploratorio detallado, registrado en el notebook *AnalisisExploratorioDatosALA*, para comprender la estructura, tipo de variables, valores nulos y distribución de los atributos del dataset proporcionado por Microsoft. Esta exploración fue esencial para diseñar una estrategia de preprocesamiento efectiva y adaptada a la naturaleza de los datos.

A partir de este análisis se definió la función `preprocess_data`, implementada en los notebooks *Entrenamiento1* y *Entrenamiento0*. Esta función incorpora la imputación de valores faltantes y preparación de los datos para su posterior uso con modelos de machine learning. Además, una vez aplicada, se complementa con la codificación categórica (One-Hot Encoding para variables nominales con pocas clases únicas y Target Encoding para categóricas ordinales o de alta cardinalidad),

Entrenamiento de modelos: Fase 1 – Baseline

En la primera etapa experimental se entrenaron tres modelos base con el objetivo de establecer un punto de referencia (“baseline”) para el rendimiento esperado sin técnicas de reducción robusta de dimensionalidad. Los modelos entrenados fueron:

1. **Regresión logística**, como primera aproximación para evaluar linealidad.
2. **Random Forest**, como modelo de tipo ensemble con capacidad para manejar datos tabulares.
3. **LightGBM**, un modelo de boosting basado en histogramas que ha demostrado un rendimiento competitivo en múltiples competencias y escenarios reales.

Los resultados mostraron que la regresión logística no logró capturar relaciones significativas en los datos, mientras que LightGBM alcanzó un AUC de 0.7079, superando a los demás y estableciendo así el baseline del proyecto.

Entrenamiento de modelos: Fase 2 – Integración de PCA clásico y robusto

En la segunda fase experimental, se integraron dos enfoques distintos de reducción de dimensionalidad:

1. **PCA clásico**, tras estandarización de variables numéricas.
2. **PCA robusto**, que incorpora:
 - Estimación robusta de la media y covarianza usando el algoritmo **Minimum Covariance Determinant (MCD)**.
 - Detección de outliers multivariantes mediante **distancia de Mahalanobis**.
 - Remoción opcional de outliers (no implementada por crash del entorno).
 - Escalado robusto basado en Σ_{robusta} .
 - PCA posterior aplicado sobre datos limpios.

Este procedimiento fue implementado en el método `preprocesar_con_mahalanobis_y_pca`.

Evaluación de modelos avanzados

Los modelos evaluados fueron:

- **LightGBM + PCA Robusto**
 - AUC: 0.6952
 - Accuracy: 0.6378
 - F1: 0.6275
- **Random Forest + PCA Robusto**
 - AUC: 0.6911
 - Accuracy: 0.6351
- **LightGBM + PCA clásico**
 - AUC: 0.7052

- Accuracy: 0.6449

Aunque el PCA robusto mostró mejoras respecto al baseline en cuanto a estabilidad frente a outliers, el PCA clásico combinado con LightGBM logró resultados casi equivalentes al modelo base, lo que sugiere que un preprocesamiento cuidadoso puede reducir la necesidad de métodos robustos si el dataset no presenta una contaminación extrema.

Nota: Realmente se hicieron dos entrenamientos para cada uno de los modelos LightGBM + PCA Robuto y Random Forest + PCA Robusto. En cada entrenamiento hay una diferencia en el preprocesamiento de los datos; sin embargo, los resultados fueron muy similares (para más detalles vea el notebook Entrenamiento1).

Referencias

- Croux, C., & Haesbroeck, G. (2000). *Principal component analysis based on robust estimators of the covariance or correlation matrix: Influence functions and efficiencies*. Biometrika, 87(3), 603–618. <https://doi.org/10.1093/biomet/87.3.603>
- David, B., & Bastin, G. (2000). *An estimator of the inverse covariance matrix and its application to ML parameter estimation in dynamical systems*. Automatica, 36(6), 877–883. [https://doi.org/10.1016/S0005-1098\(00\)00010-1](https://doi.org/10.1016/S0005-1098(00)00010-1)
- Deng, X., & Tsui, K. L. (2013). *Penalized likelihood approach to sparse portfolio selection*. Statistica Sinica, 23(1), 119–137.
- Hubert, M., Rousseeuw, P. J., & Vanden Branden, K. (2005). *ROBPCA: A new approach to robust principal component analysis*. Technometrics, 47(1), 64–79. <https://doi.org/10.1198/004017004000000563>
- Kang, X., Lian, J., & Deng, X. (2021). *On Block Cholesky Decomposition for Sparse Inverse Covariance Estimation*. Statistica Sinica (versión aceptada). <https://doi.org/10.5705/ss.202020.0211>
- Ledoit, O., & Wolf, M. (2004). *A well-conditioned estimator for large-dimensional covariance matrices*. Journal of Multivariate Analysis, 88(2), 365–411. [https://doi.org/10.1016/S0047-259X\(03\)00096-4](https://doi.org/10.1016/S0047-259X(03)00096-4)
- Maronna, R. A., Martin, R. D., & Yohai, V. J. (2006). *Robust Statistics: Theory and Methods*. Wiley.
- Nirpy Research. (2021). *Robust PCA: How to deal with outliers in dimensionality reduction?* Disponible en: <https://nirpyresearch.com/robust-pca/>
- Pourahmadi, M. (1999). *Modified Cholesky Decomposition and Covariance Estimation*. Biometrika, 86(3), 587–602. <https://doi.org/10.1093/biomet/86.3.587>
- Rousseeuw, P. J. (1984). *Least median of squares regression*. Journal of the American Statistical Association, 79(388), 871–880. <https://doi.org/10.2307/2288718>
- Rousseeuw, P. J., & Van Driessen, K. (1999). *A Fast Algorithm for the Minimum Covariance Determinant Estimator*. Technometrics, 41(3), 212–223. <https://doi.org/10.1080/00401706.1999.10485670>
- scikit-learn developers. (2024). *MinCovDet* — *scikit-learn 1.4.1 documentation*. <https://scikit-learn.org/stable/modules/generated/sklearn.covariance.MinCovDet.html>

- statisticseasily.com. (n.d.). *What is Minimum Covariance Determinant (MCD)?* <https://statisticseasily.com/glossario/what-is-minimum-covariance-determinant/>
- Yuan, M. (2007). *High dimensional inverse covariance matrix estimation via linear programming*. Journal of Machine Learning Research, 9, 2261–2289. <https://www.jmlr.org/papers/volume9/yuan08a/yuan08a.pdf>